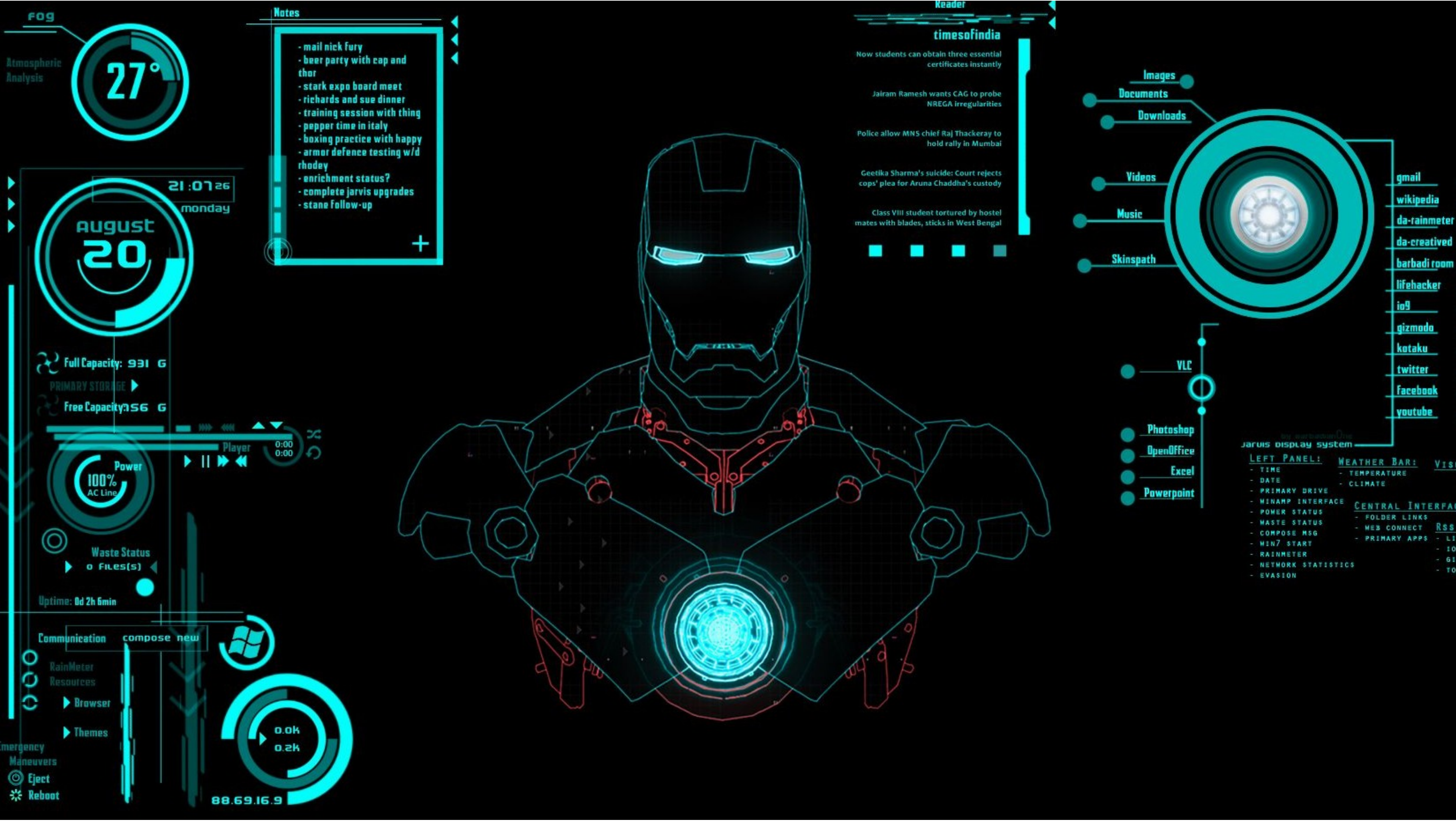


06.Transformações Geométricas

Prof. Alexandre Krohn

Roteiro

- Transformações Geométricas
- Translação
- Escala
- Rotação
- Transformações Lineares

Transformações Geométricas

- Relembrando, o processo de Visualização 2D é composto por:
  - Instanciação
  - Recorte
  - Mapeamento

Transformações Geométricas

- No processo de ***Instanciação***, após criar as instâncias dos objetos que serão desenhados, suas **características podem ser modificadas**, através de processos conhecidos como transformações geométricas.

Transformações Geométricas

- Transformações geométricas são funções utilizadas para mapear pontos do espaço em outros pontos

Transformações Geométricas

- Podem ser ajustadas, através de transformações, as seguintes características:
  - Posição
  - Escala
  - Inclinação

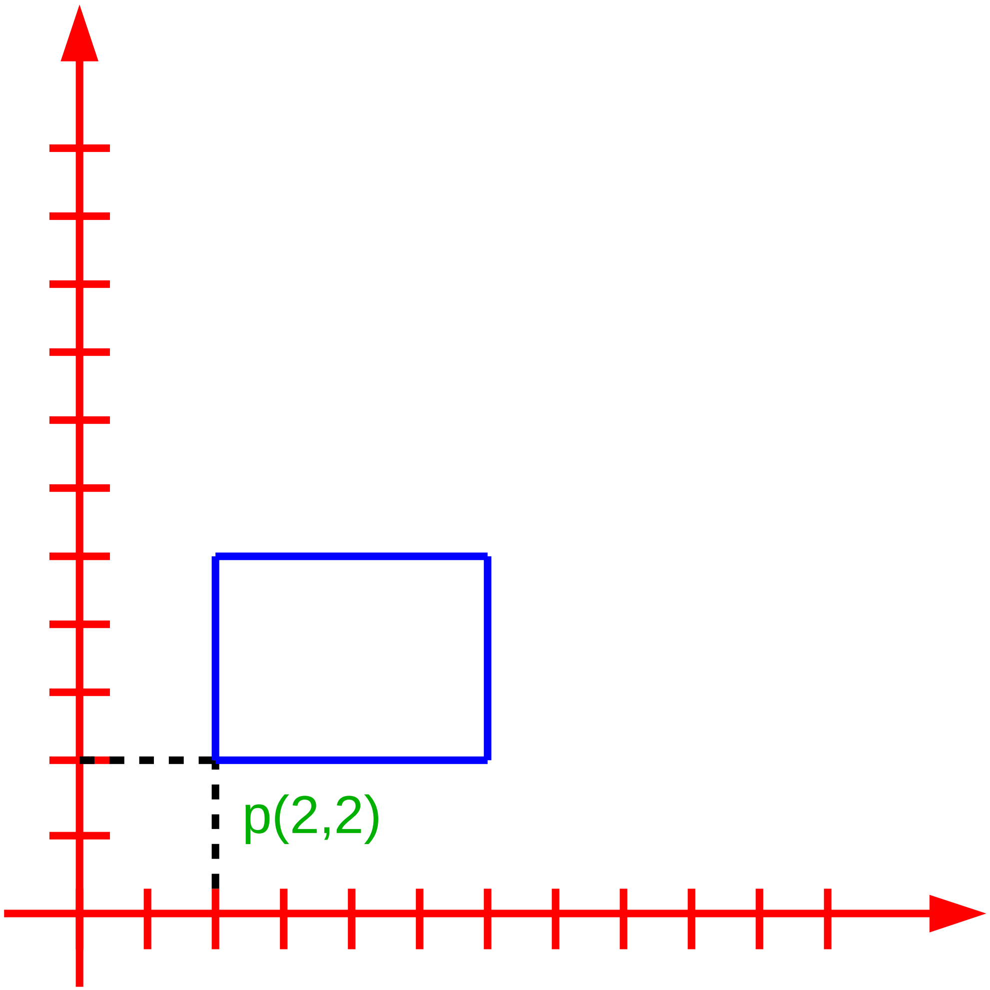
Roteiro

- Transformações Geométricas
- Translação
- Escala
- Rotação
- Transformações Lineares

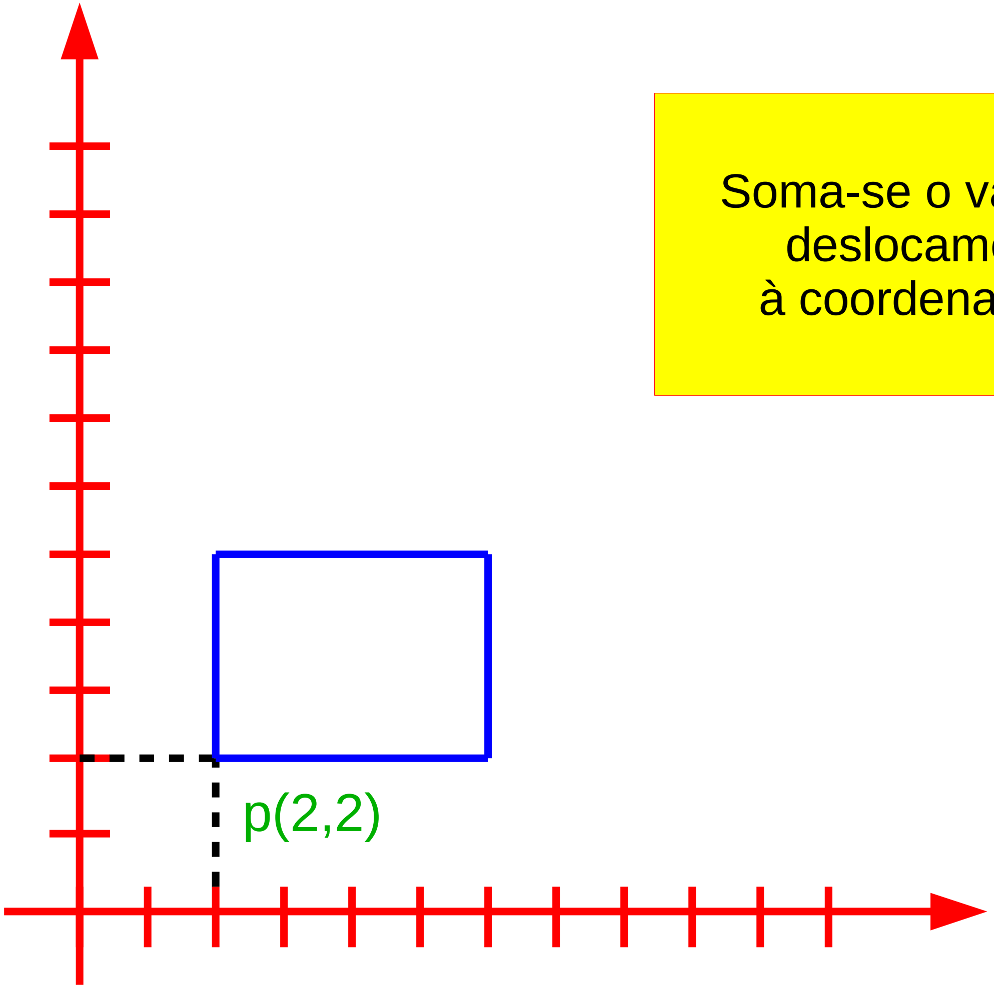
Translações

- Translações mudam a posição do polígono que representa uma figura na área desenhada
- São operações simples, resultantes da adição de valores de deslocamento às coordenadas de **cada** ponto dos polígonos

Translações : Polígono Original

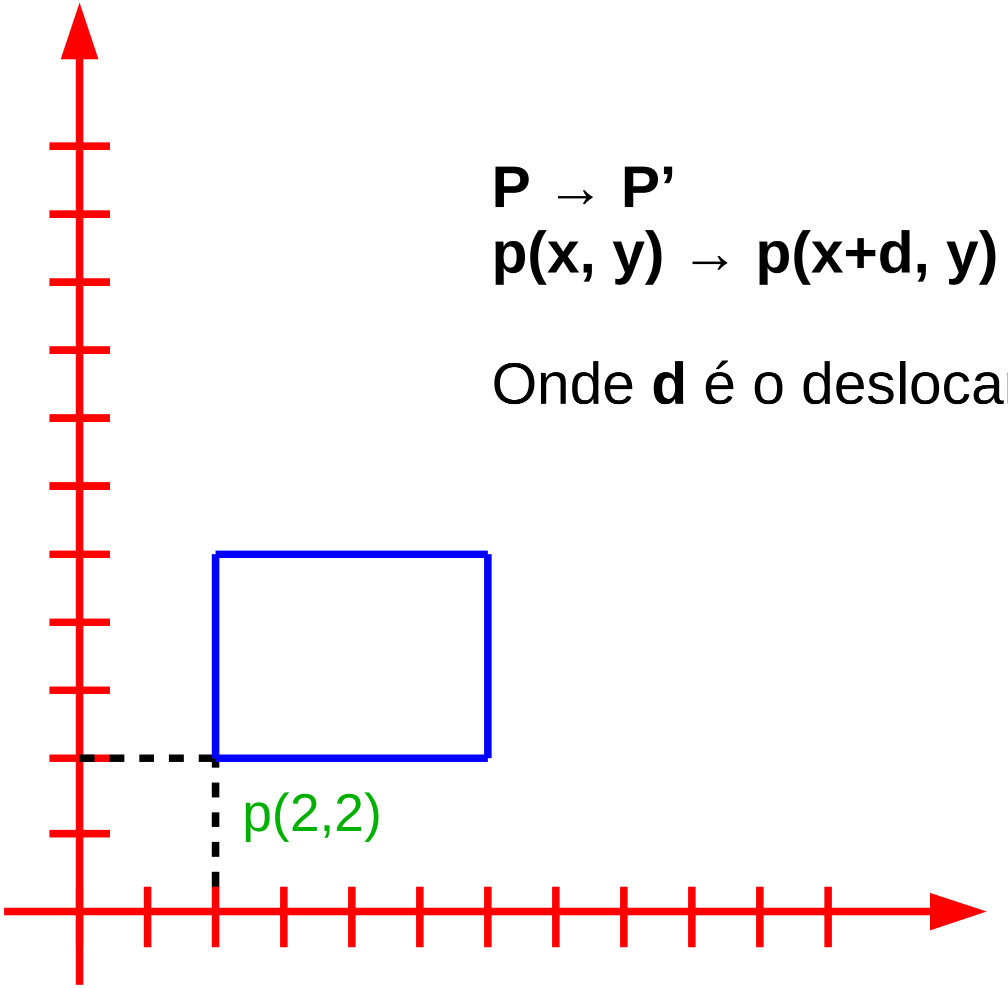


Translação horizontal



Soma-se o valor do deslocamento à coordenada **X**

Translação horizontal



P → P'
p(x, y) → p(x+d, y)

Onde **d** é o deslocamento horizontal

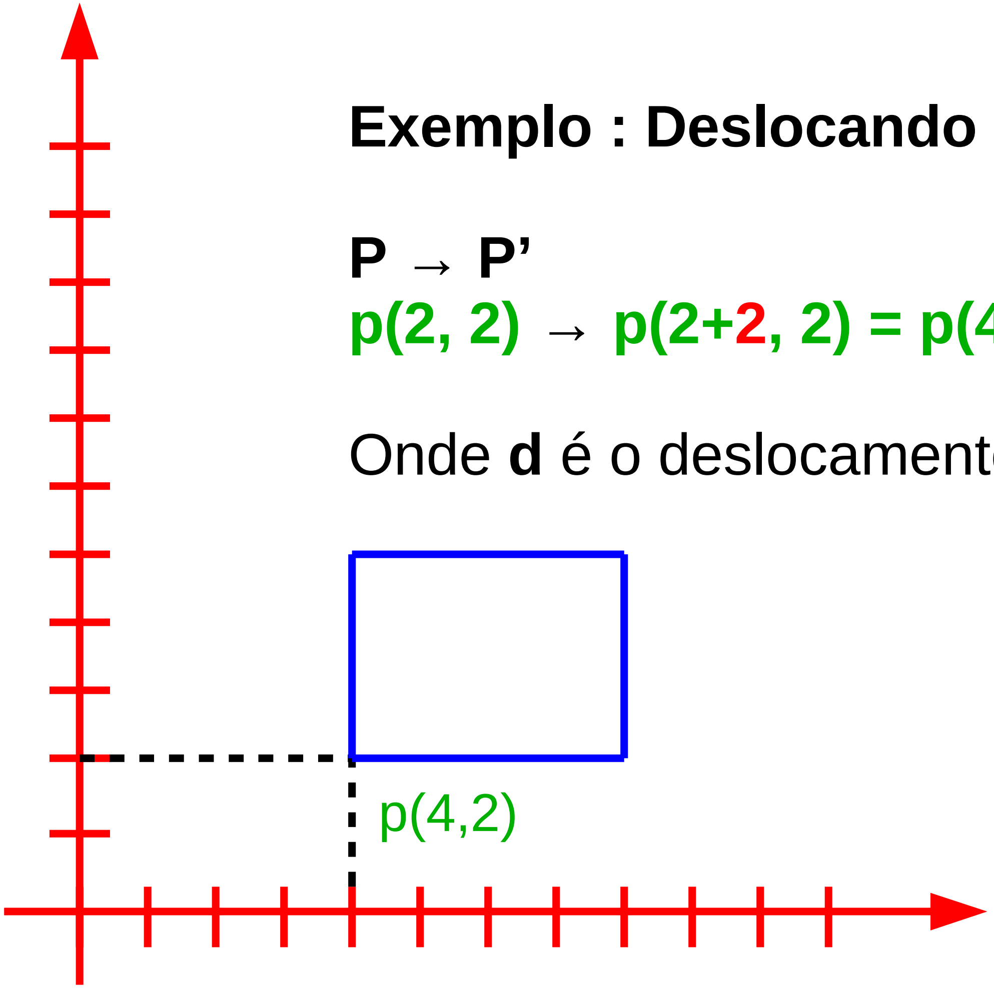
Translação horizontal

Exemplo : Deslocando 2 unidades

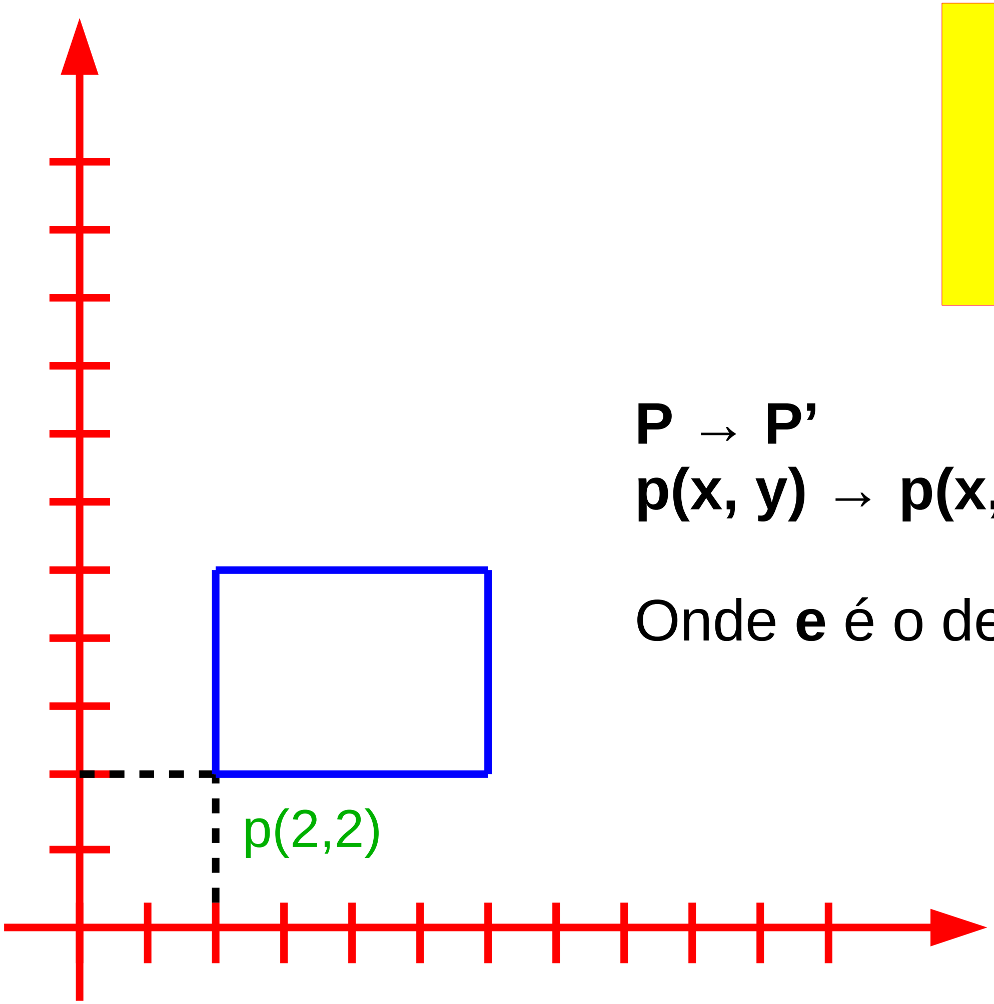
P → P'

p(2, 2) → p(2+2, 2) = p(4,2)

Onde **d** é o deslocamento horizontal



Translação vertical

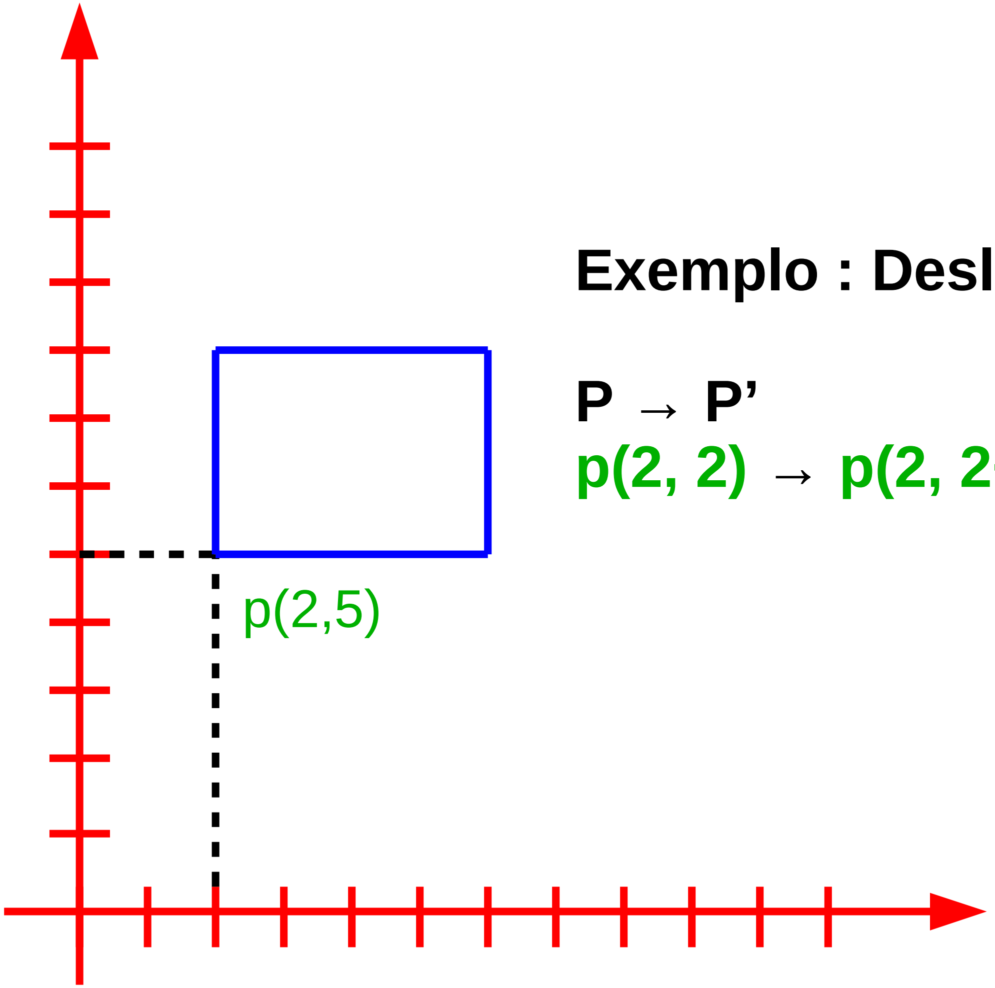


Soma-se o valor do deslocamento à coordenada **Y**

P → P'
p(x, y) → p(x, y+e)

Onde **e** é o deslocamento vertical

Translação vertical



Exemplo : Deslocando 3 unidades

P → P'

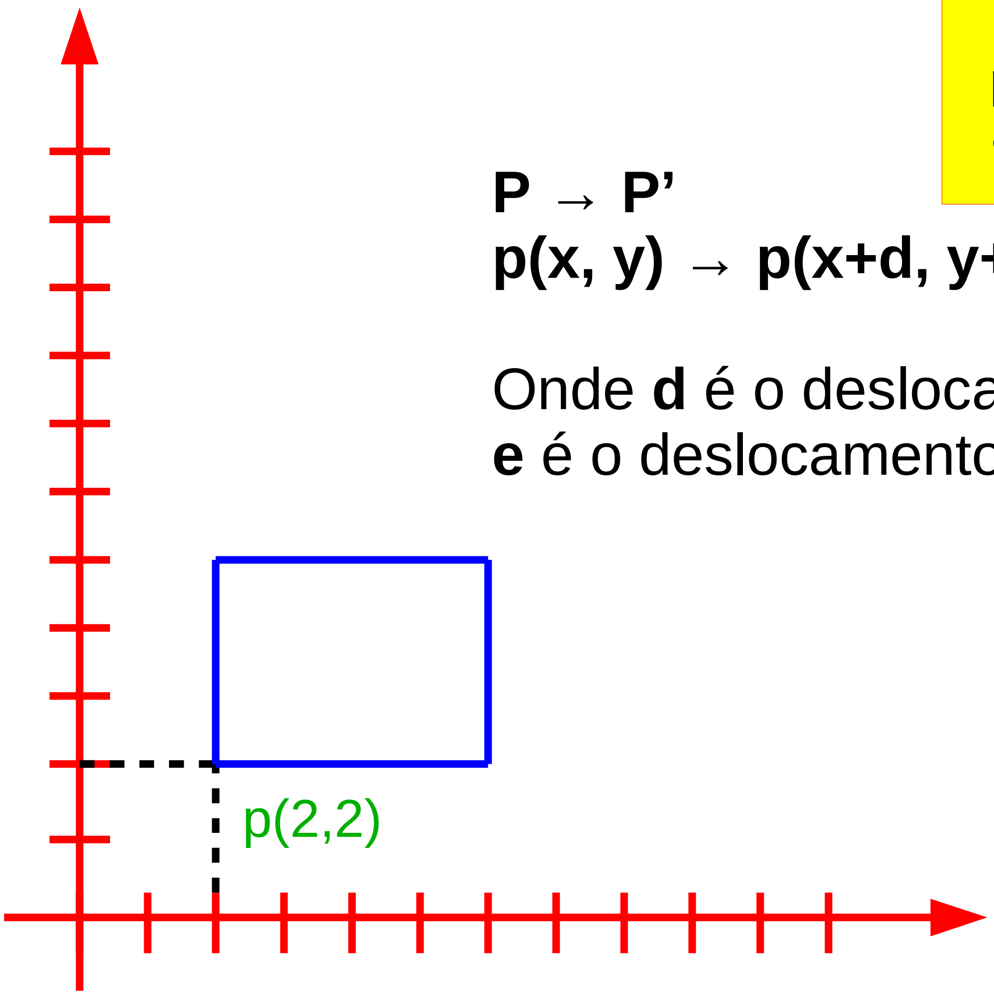
p(2, 2) → p(2, 2+3) = p(2,5)

Translação horizontal e vertical

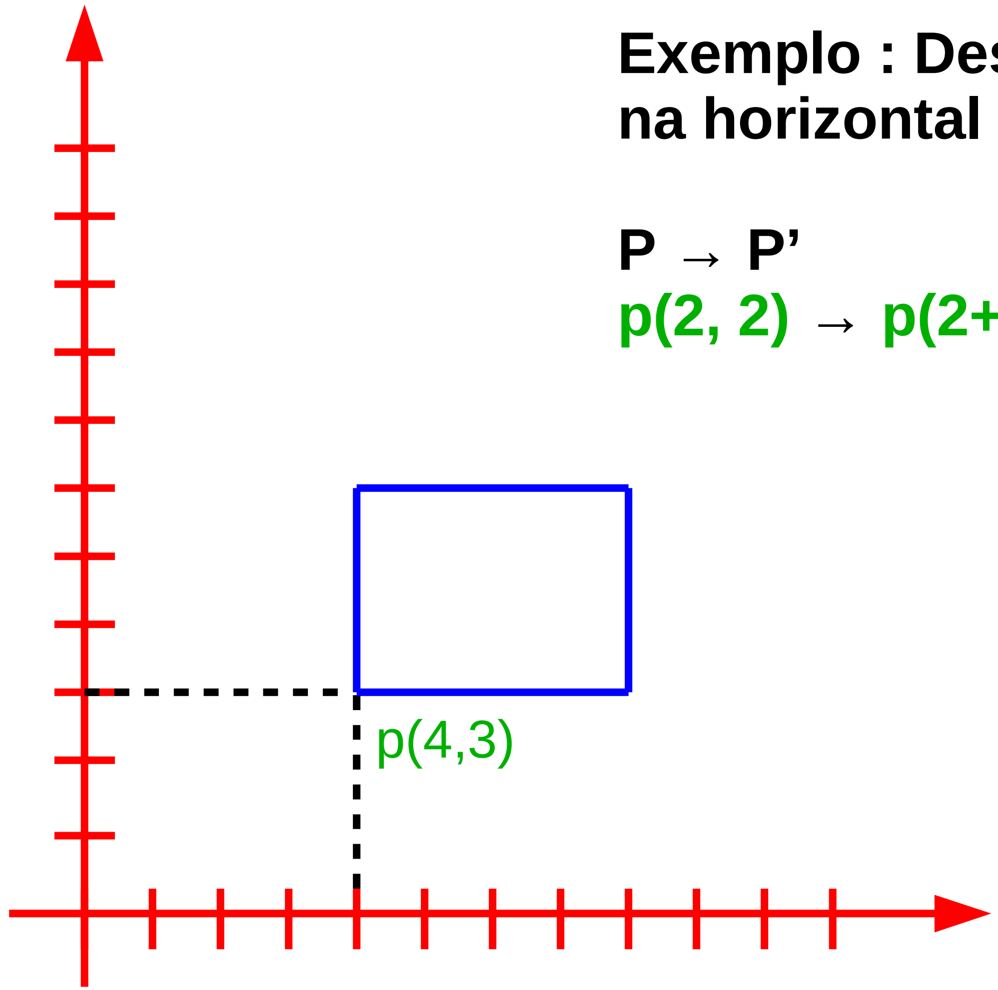
Soma-se o valores de deslocamento, que podem ser diferentes, à coordenadas **X** e **Y**

P → P'
p(x, y) → p(x+d, y+e)

Onde **d** é o deslocamento horizontal e **e** é o deslocamento vertical



Translação horizontal e vertical



Exemplo : Deslocando 2 unidades na horizontal e 1 unidade na vertical

P → P'

p(2, 2) → p(2+2, 2+1) = p(4,3)

Translações

- É importante salientar que a operação de translação compreende adicionar valores à **TODOS** os pontos do polígono desenhado
- Translações no sentido direita para esquerda e de cima para baixo podem ser obtidas somando-se valores negativos às coordenadas

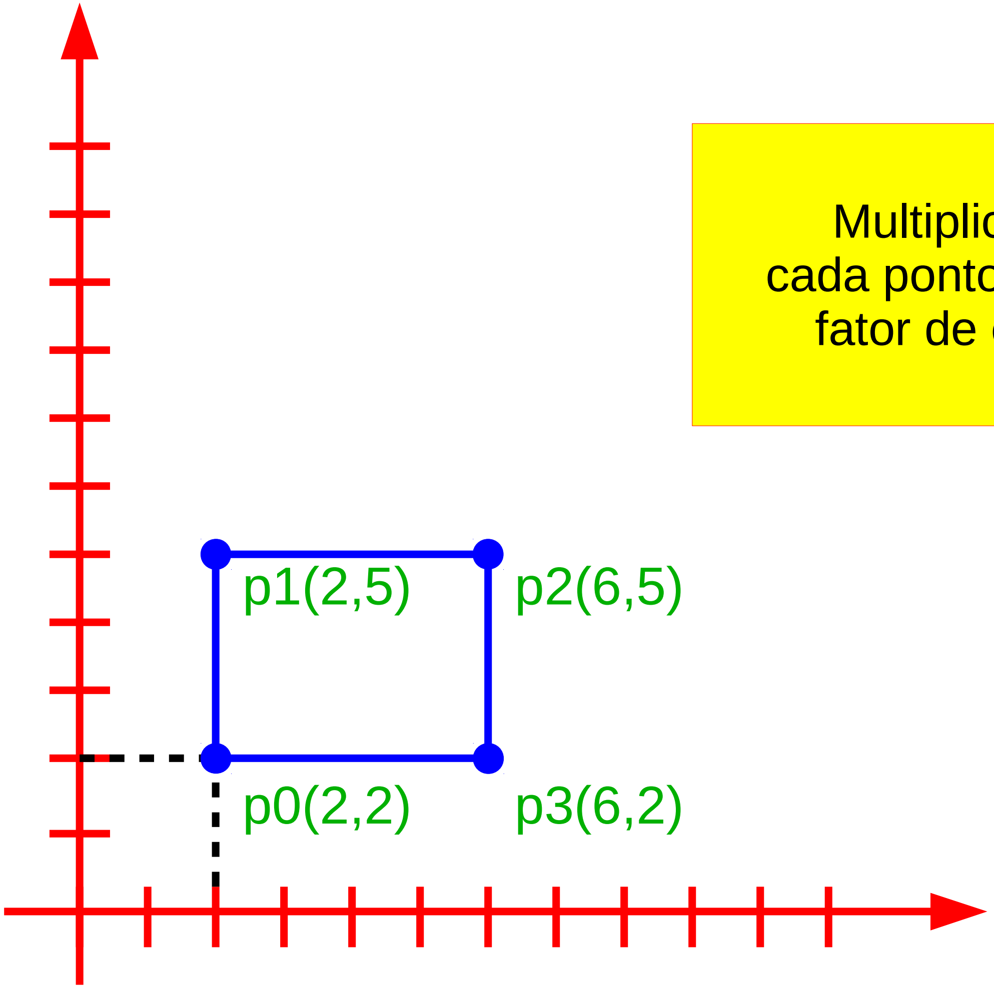
Roteiro

- Transformações Geométricas
- Translação
- Escala
- Rotação
- Transformações Lineares

Escala

- Transformações de escala alteram as dimensões dos polígonos sendo apresentados.
- Elas são obtidas multiplicando-se coordenadas por um **fator de escala**
  - Fatores maiores do que 1 aumentam o polígono
  - Fatores menores do que 1 reduzem seu tamanho

Escala

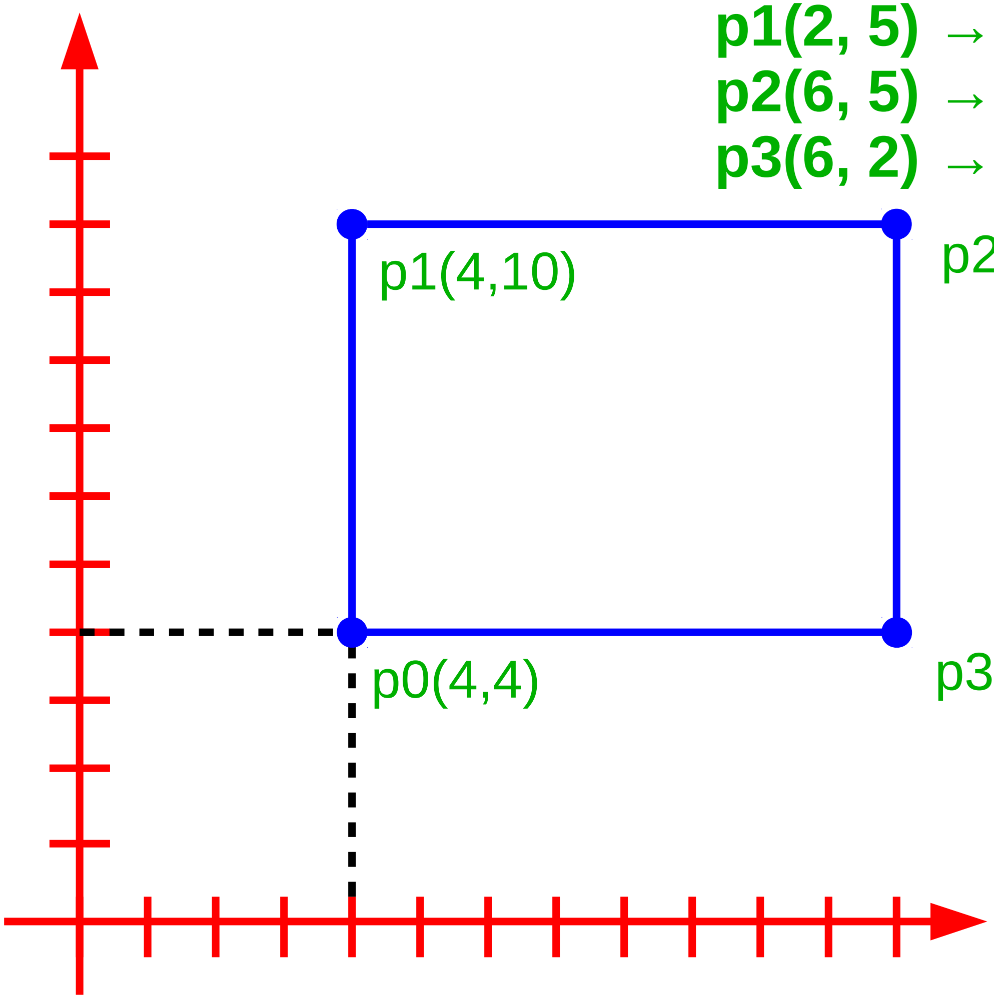


Multiplica-se
cada ponto por um
fator de escala

Escala

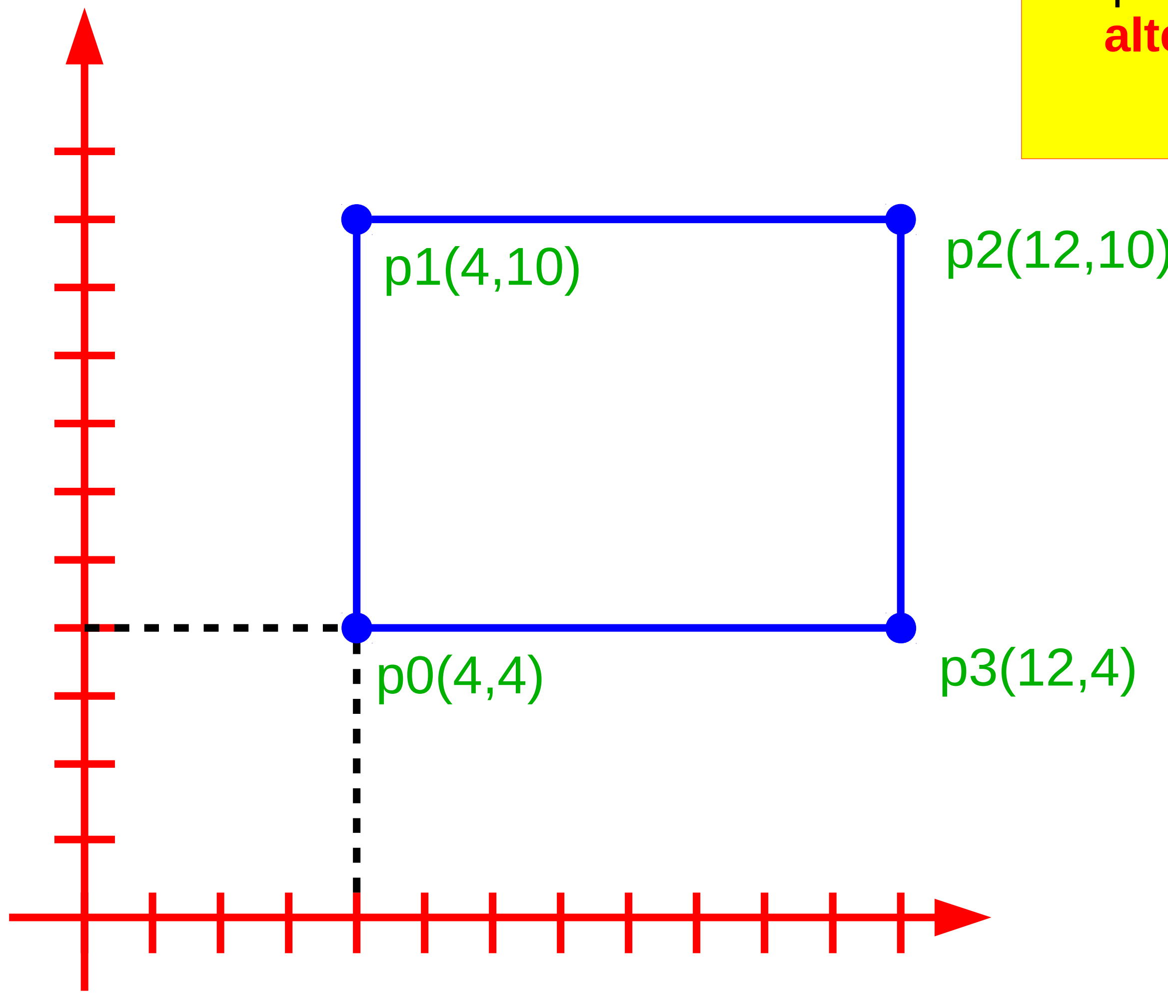
Exemplo : Multiplicando-se cada coordenada por 2

P → P'
p0(2, 2) → p0(2*2, 2*2) = p0(4,4)
p1(2, 5) → p1(2*2, 5*2) = p1(4,10)
p2(6, 5) → p2(6*2, 5*2) = p2(12,10)
p3(6, 2) → p3(6*2, 2*2) = p3(12,4)

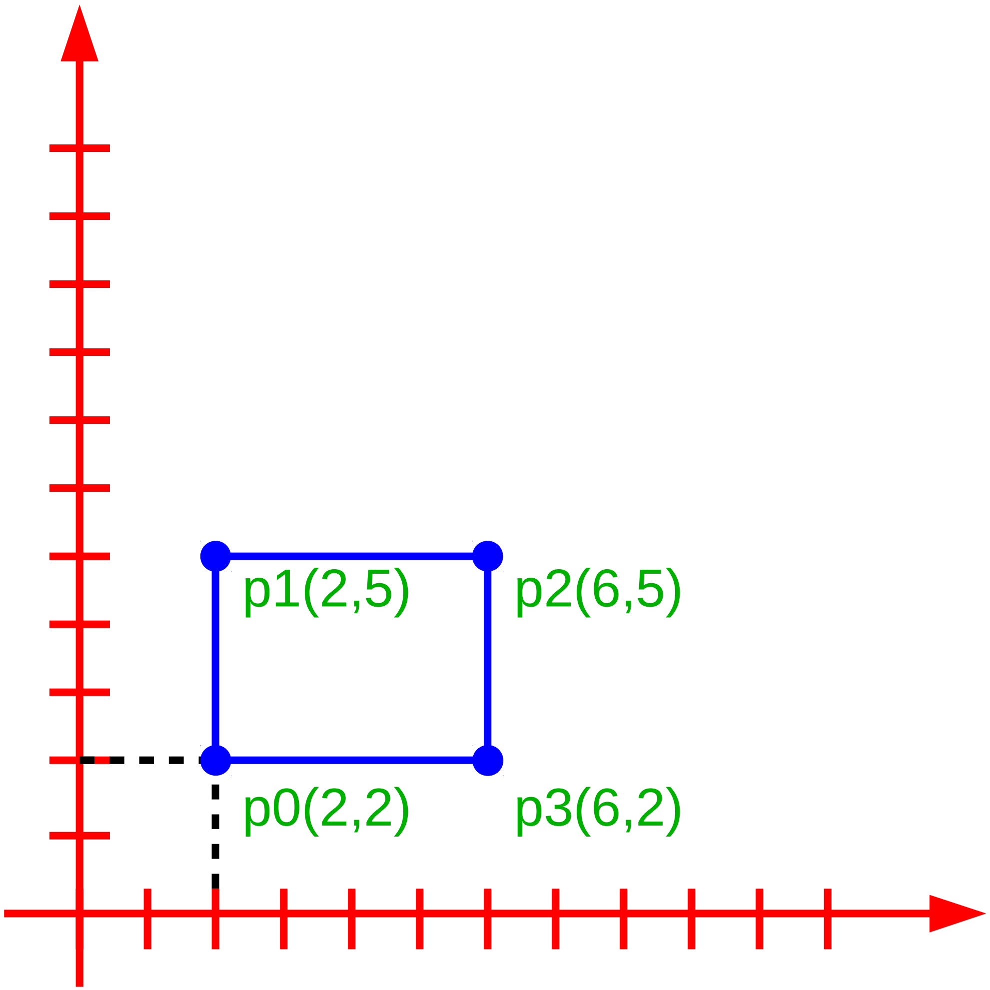


Escala

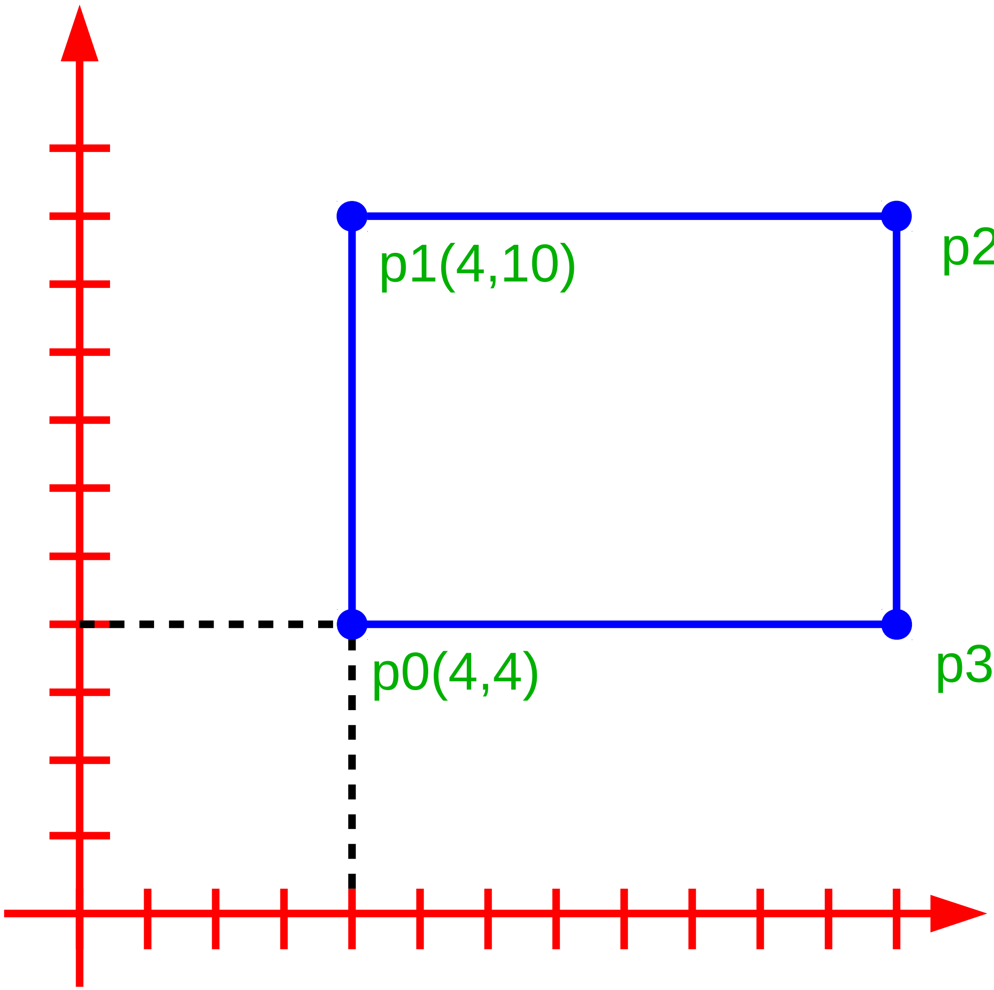
Perceba-se que a operação de escala **alterou a posição** do polígono



Escala (Posição Original)



Escala (Nova Posição)

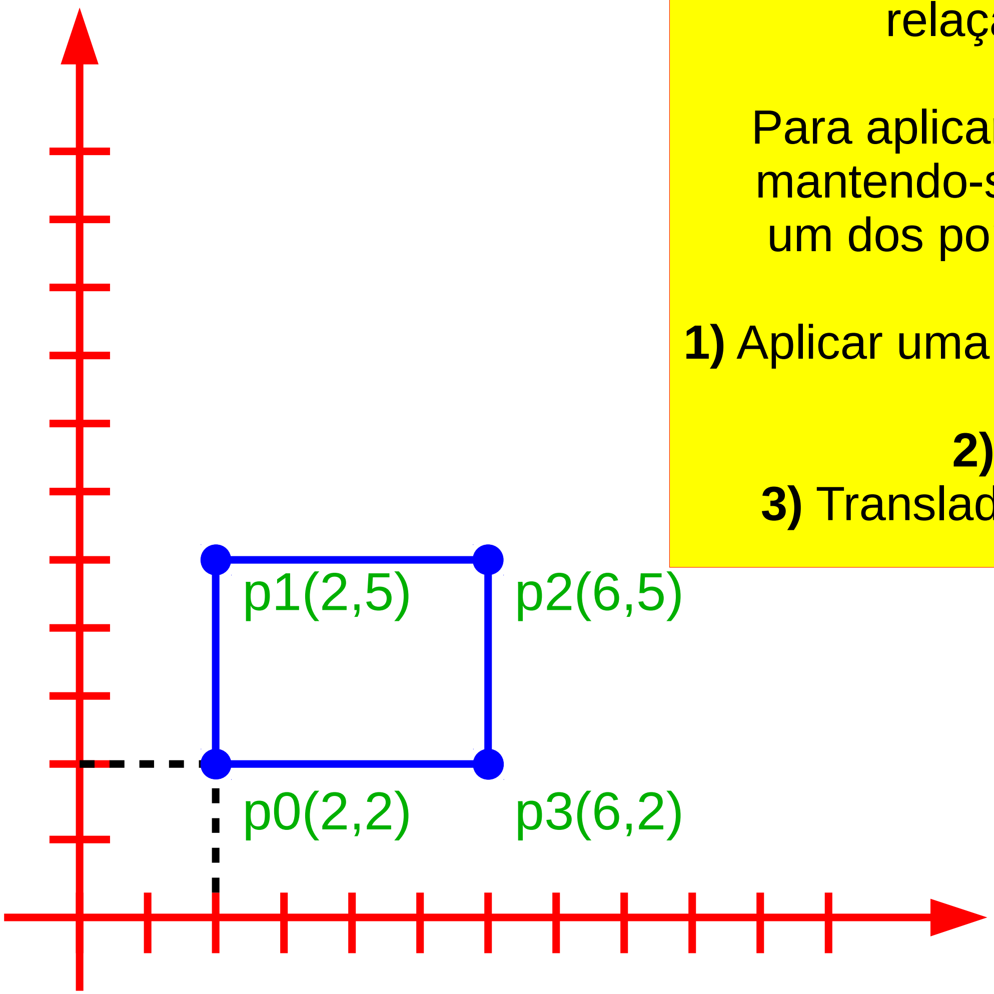


Escala em torno de um ponto árbitrário

A operação de escala é sempre feita em relação à origem **PO(0,0)**.

Para aplicar uma operação de escala, mantendo-se a posição de original de um dos pontos do polígono, deve-se:

1) Aplicar uma translação do ponto escolhido até a origem,
2) Aplicar a escala,
3) Transladá-lo até sua posição inicial

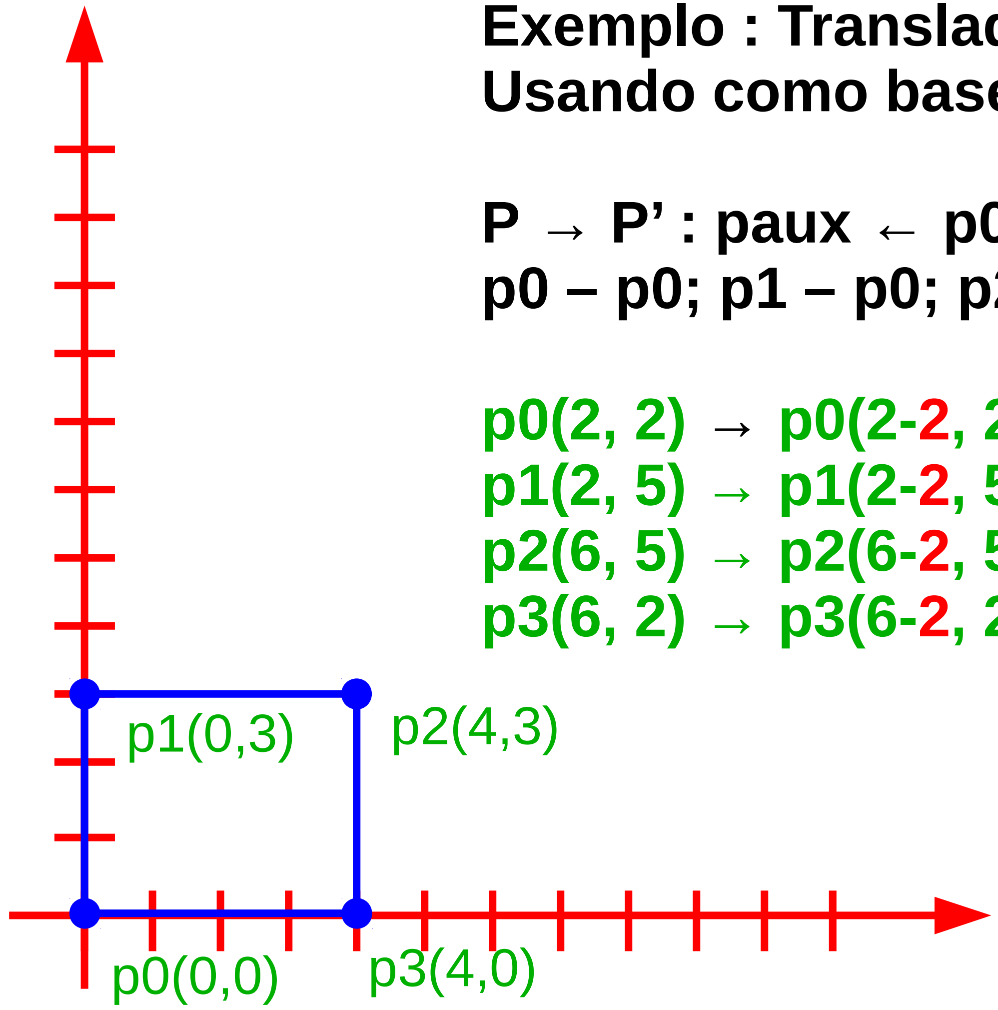


Escala (Passo 1)

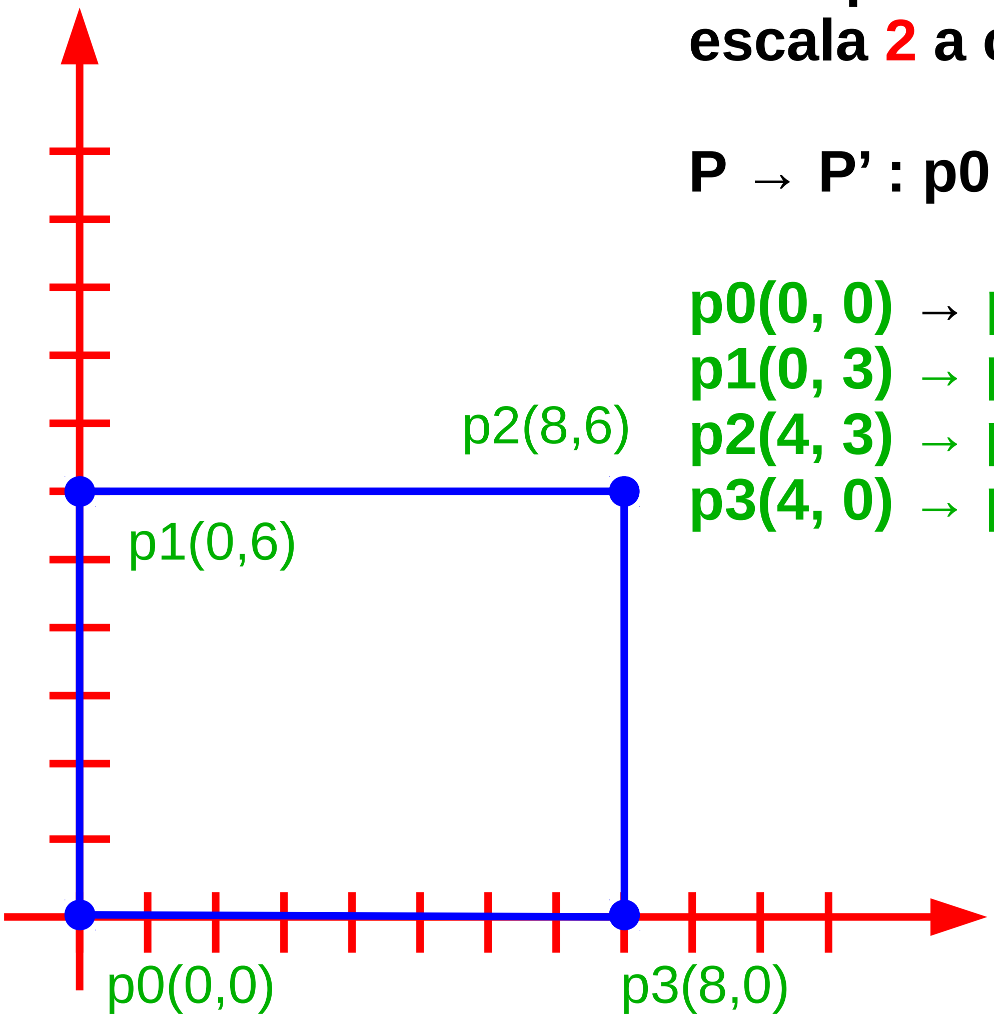
Exemplo : Transladar para origem,
Usando como base o ponto p0(2,2):

P → P' : paux ← p0 → paux(2,2)
p0 – p0; p1 – p0; p2 – p0; p3 -p0

p0(2, 2) → p0(2-2, 2-2) = p0(0,0)
p1(2, 5) → p1(2-2, 5-2) = p1(0,3)
p2(6, 5) → p2(6-2, 5-2) = p2(4,3)
p3(6, 2) → p3(6-2, 2-2) = p3(4,0)



Escala (Passo 2)



Exemplo : Aplicar o fator de escala 2 a cada ponto:

P → P' : p0 * 2; p1 * 2; p2 * 2; p3 * 2

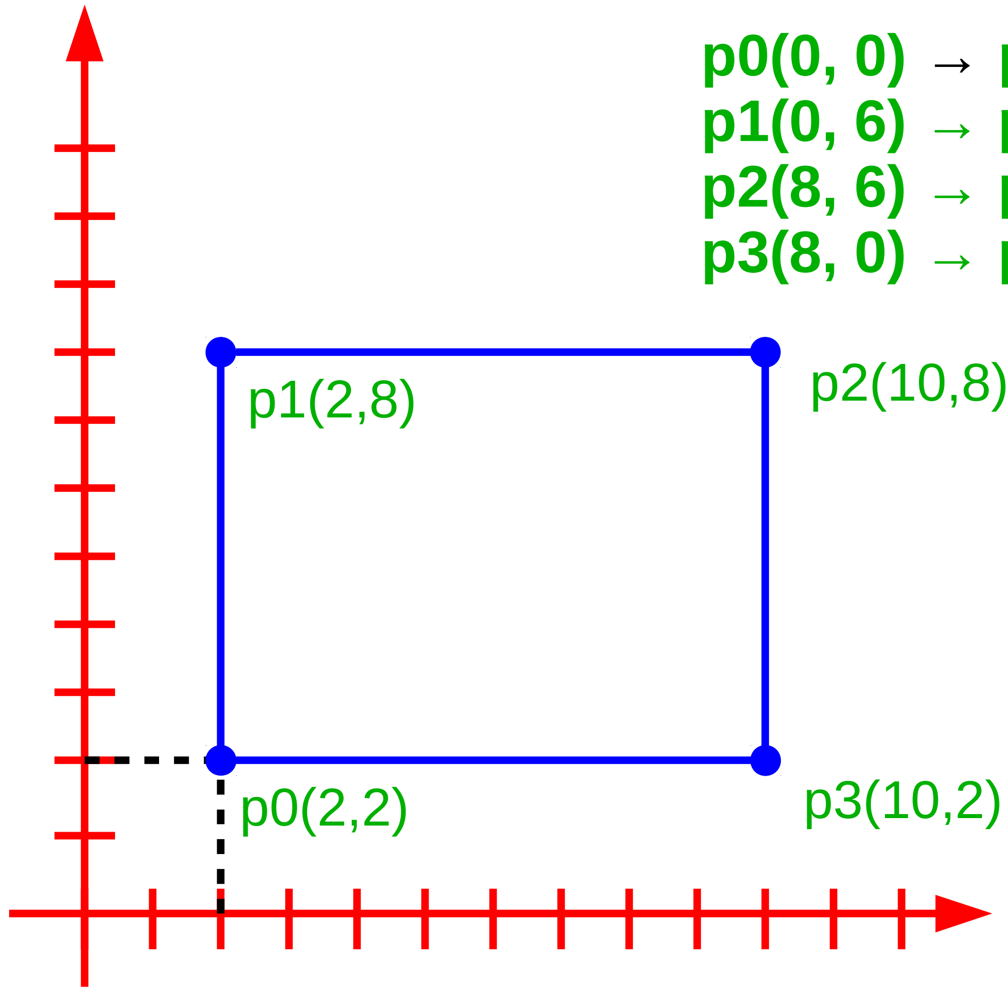
p0(0, 0) → p0(0*2, 0*2) = p0(0,0)
p1(0, 3) → p1(0*2, 3*2) = p1(0,6)
p2(4, 3) → p2(4*2, 3*2) = p2(8,6)
p3(4, 0) → p3(4*2, 0*2) = p3(8,0)

Escala (Passo 3)

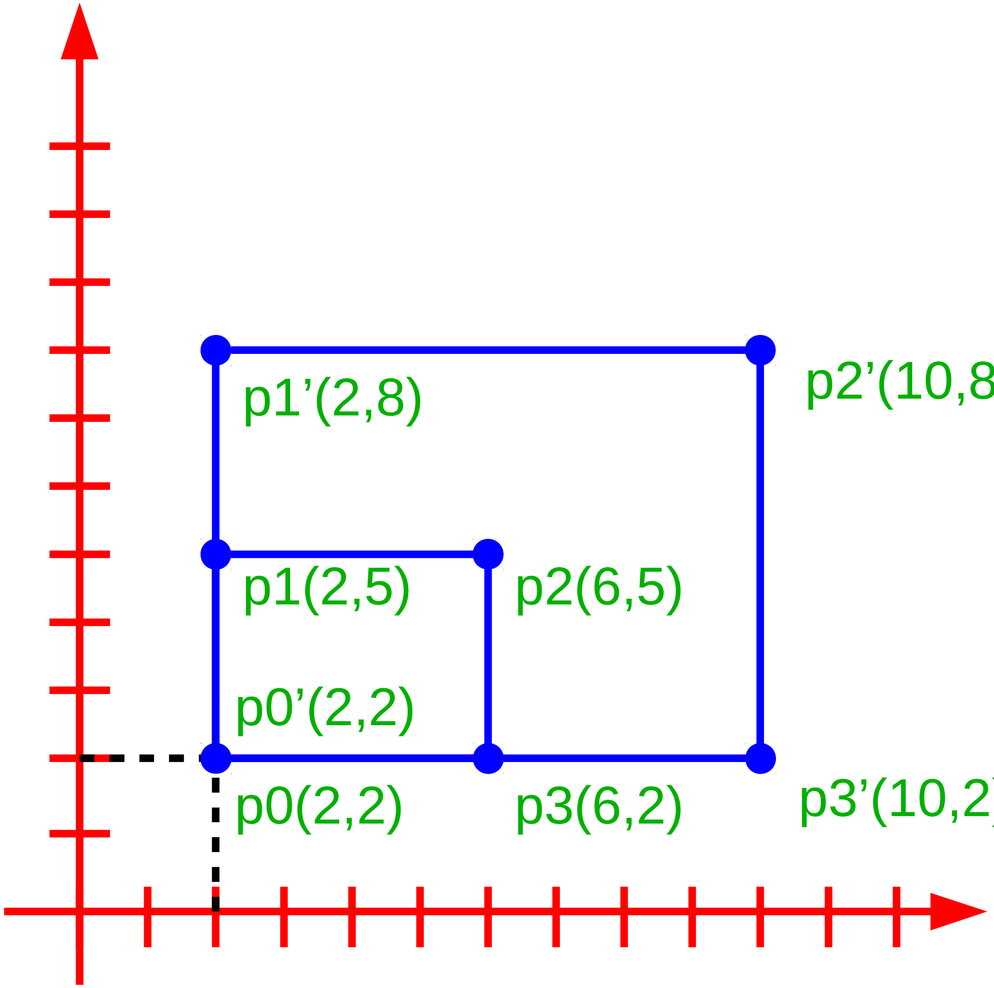
Exemplo : Transladar para o ponto inicial paux(2,2):

P → P' : p0 + paux; p1 + paux; p2 + paux; p3 + paux

p0(0, 0) → p0(0+2, 0+2) = p0(2,2)
p1(0, 6) → p1(0+2, 6+2) = p1(2,8)
p2(8, 6) → p2(8+2, 6+2) = p2(10,8)
p3(8, 0) → p3(8+2, 0+2) = p3(10,2)

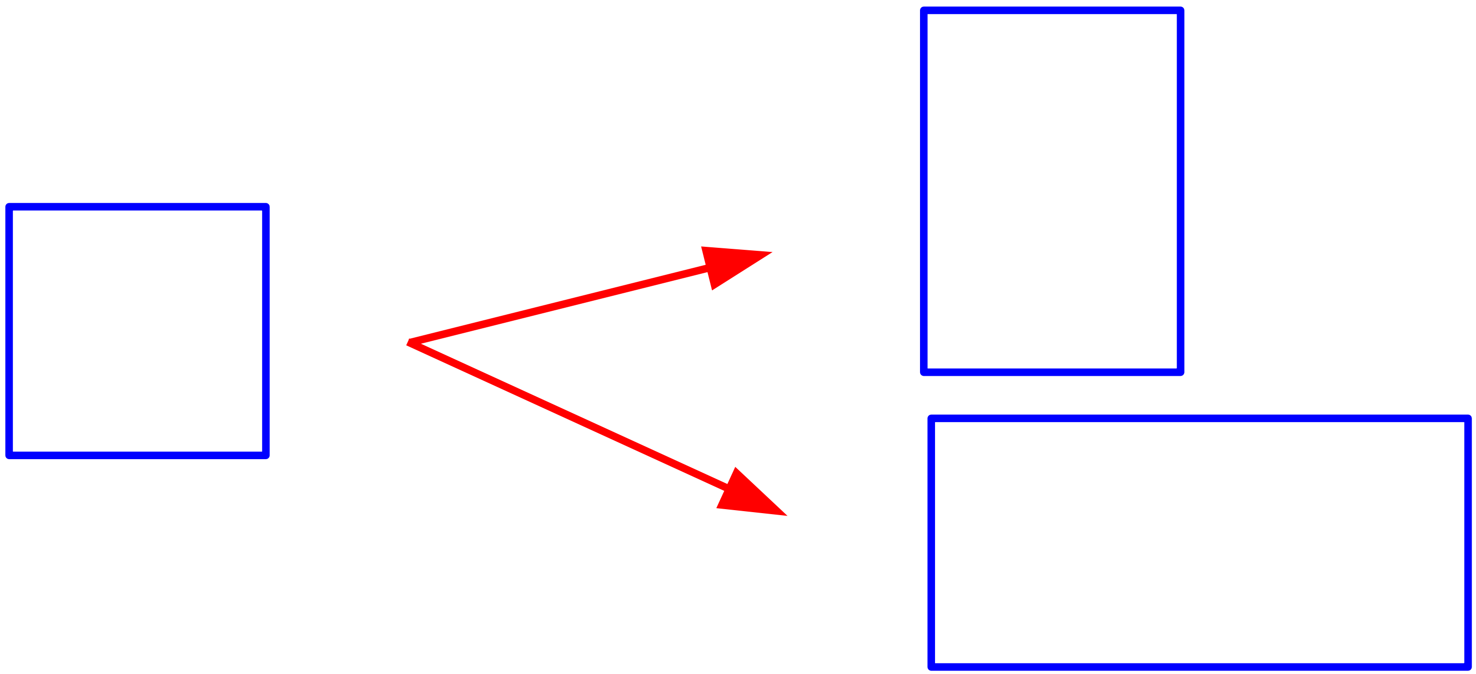


Escala :
Comparando com o original



Escala

- Se a escala aplicada nos eixos x e y for **diferente**, ocorrerá a **deformação** do objeto!



Roteiro

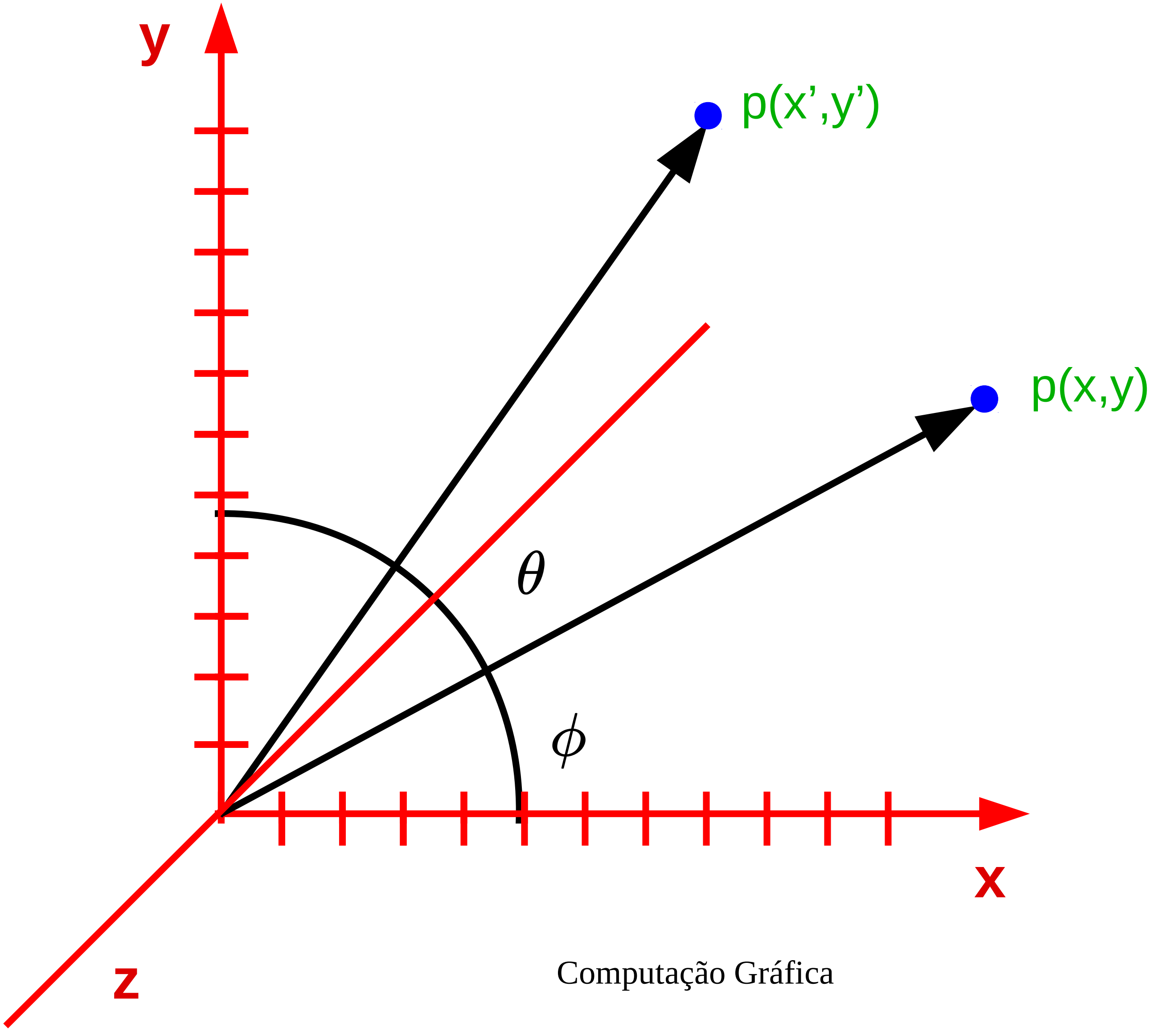
- Transformações Geométricas
- Translação
- Escala
- Rotação
- Transformações Lineares

Rotação

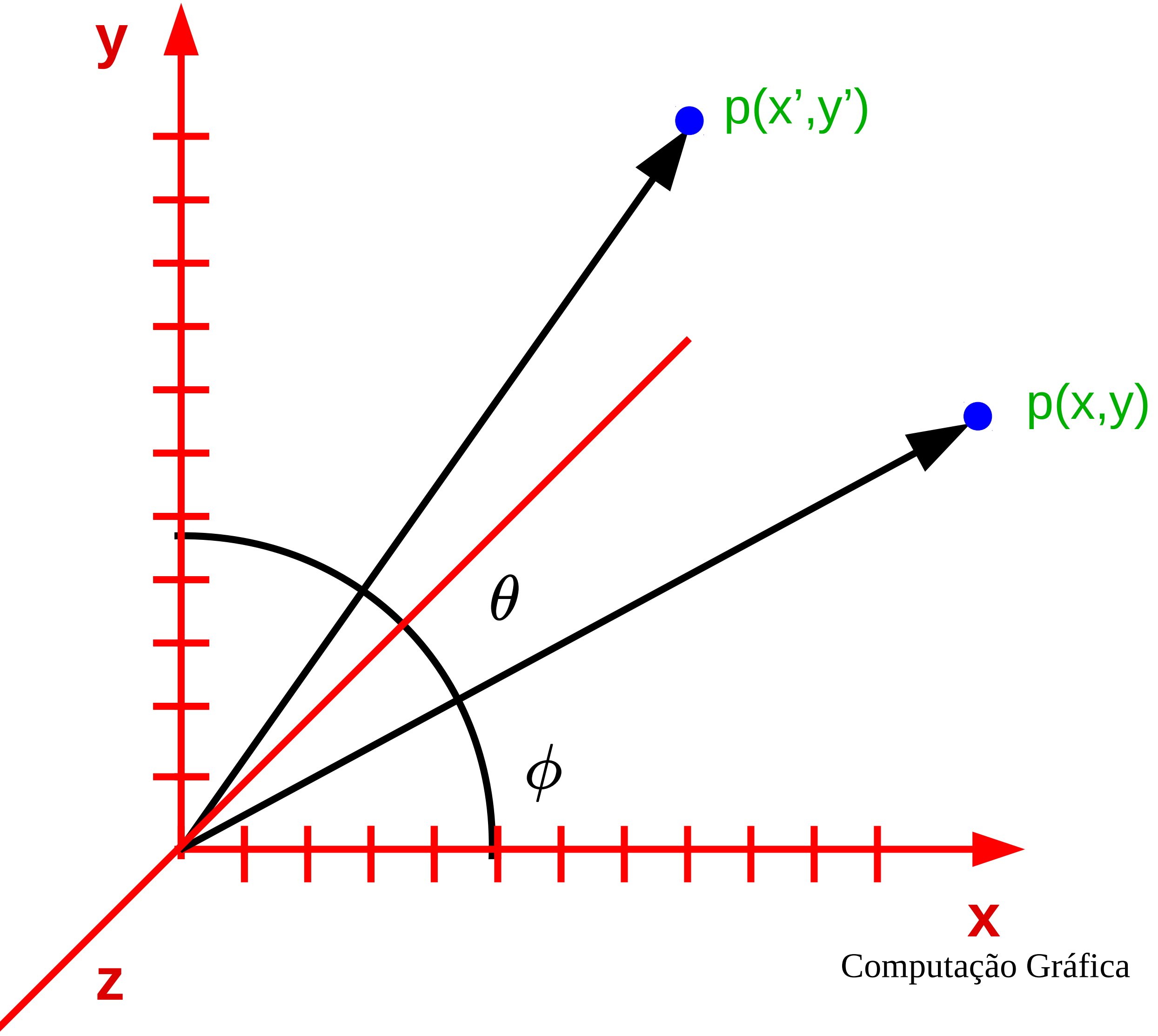
- Na operação de rotação, altera-se o ângulo, ou seja, a inclinação no qual um objeto é posicionado na tela.
- Objetos normalmente são posicionados paralelamente ao eixo x, mas esse ângulo pode ser alterado

Rotação

A operação de rotação
é realizada em torno do eixo **z**,

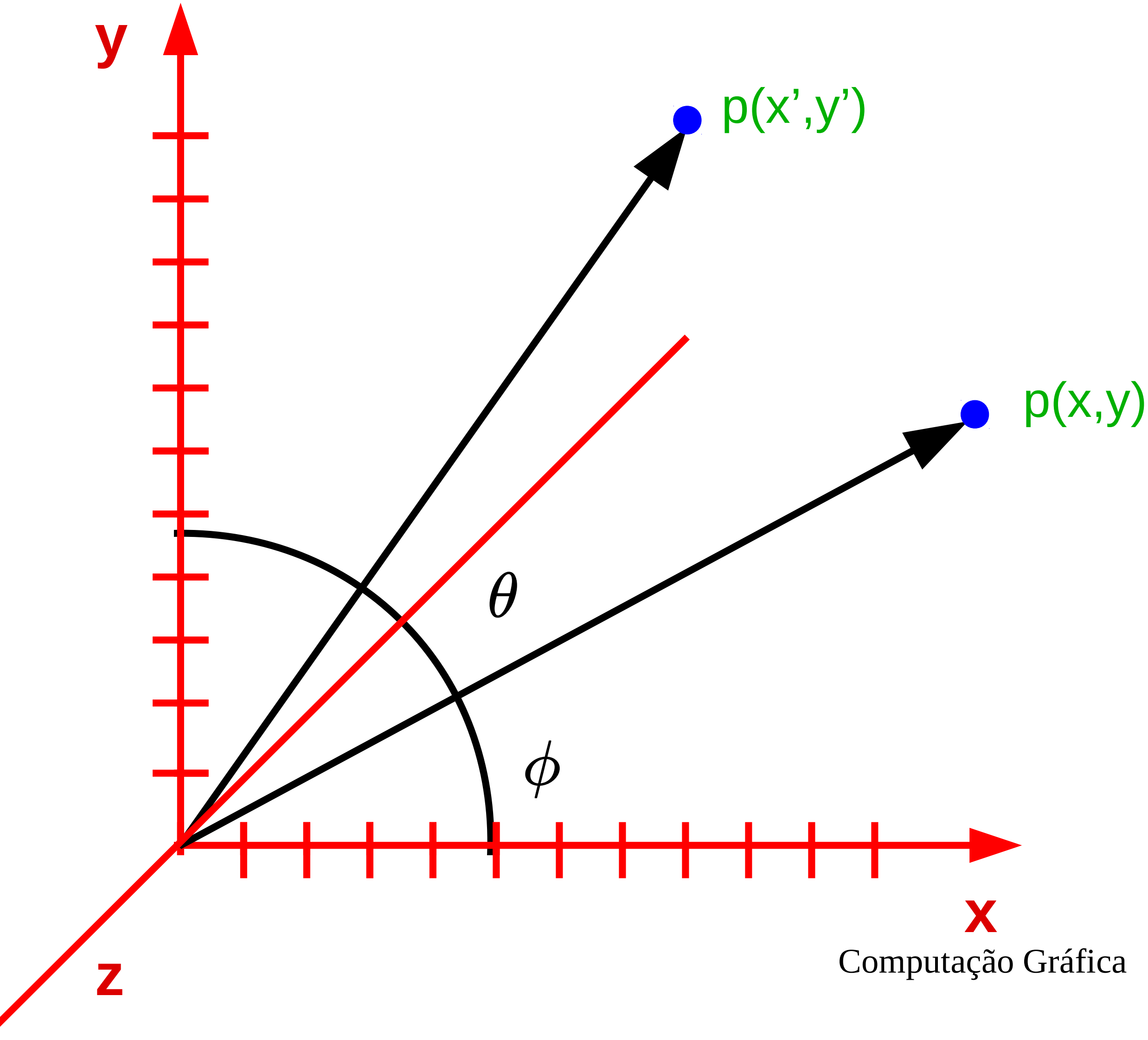


Rotação



$x = p \cos(\phi)$
$y = p \sin(\phi)$
$z = z$

Rotação



$x = p\cos(\phi)$
$y = p\sin(\phi)$
$z = z$

$x' = p\cos(\theta + \phi)$
$y' = p\sin(\theta + \phi)$
$z' = z$

Rotação

$x' = p\cos(\theta + \phi)$
$y' = p\sin(\theta + \phi)$
$z' = z$

Reescrevendo :

$x' = p\cos\theta\ \ p\cos\theta - p\sin\theta\ \ p\sin\theta = x\cos\theta - y\sin\theta$
$y' = p\cos\theta\ \ p\sin\theta - p\sin\theta\ \ p\cos\theta = x\sin\theta + y\cos\theta$

Logo :

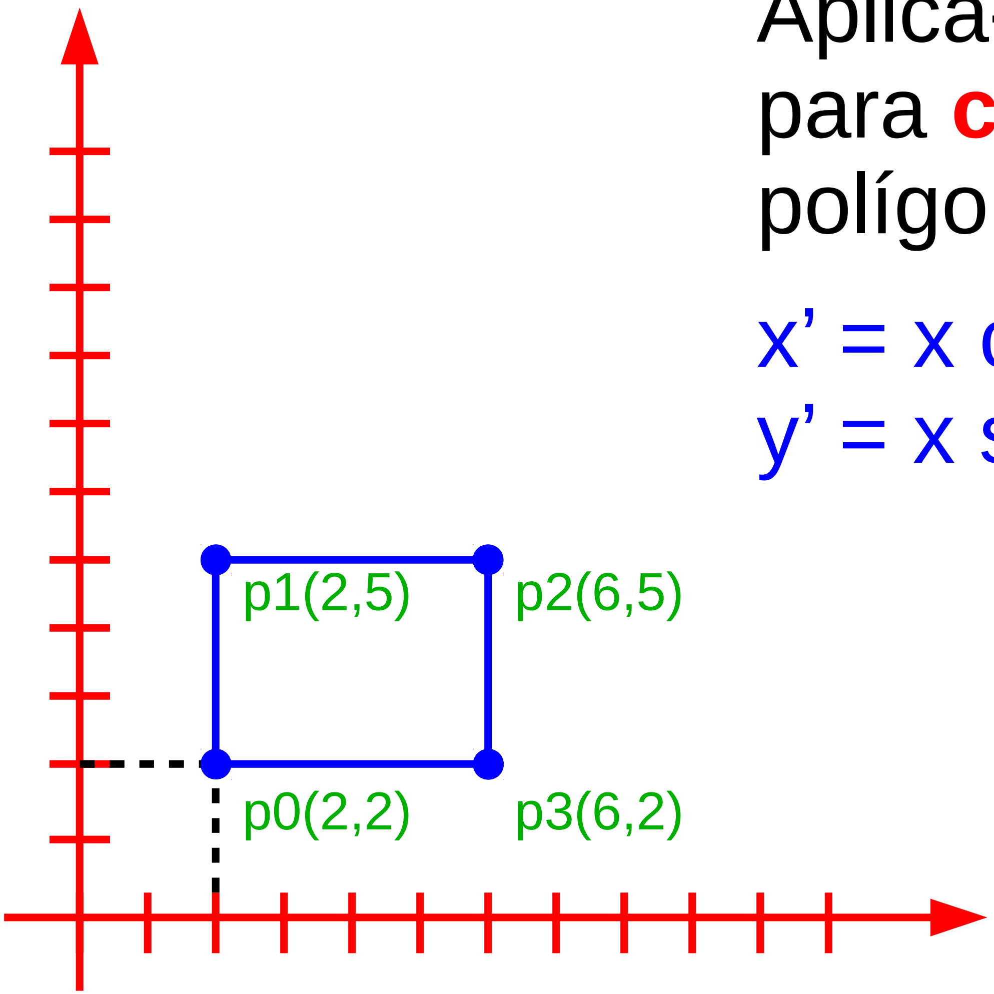
$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

Rotação : Radianos

- Lembre-se que na maioria das linguagens de programação, as funções trigonométricas recebem os ângulos em radianos. Para fazer a conversão, utilize:

```
double toRadians(double degrees)
{
    return degrees * 0.017453293;
}
```

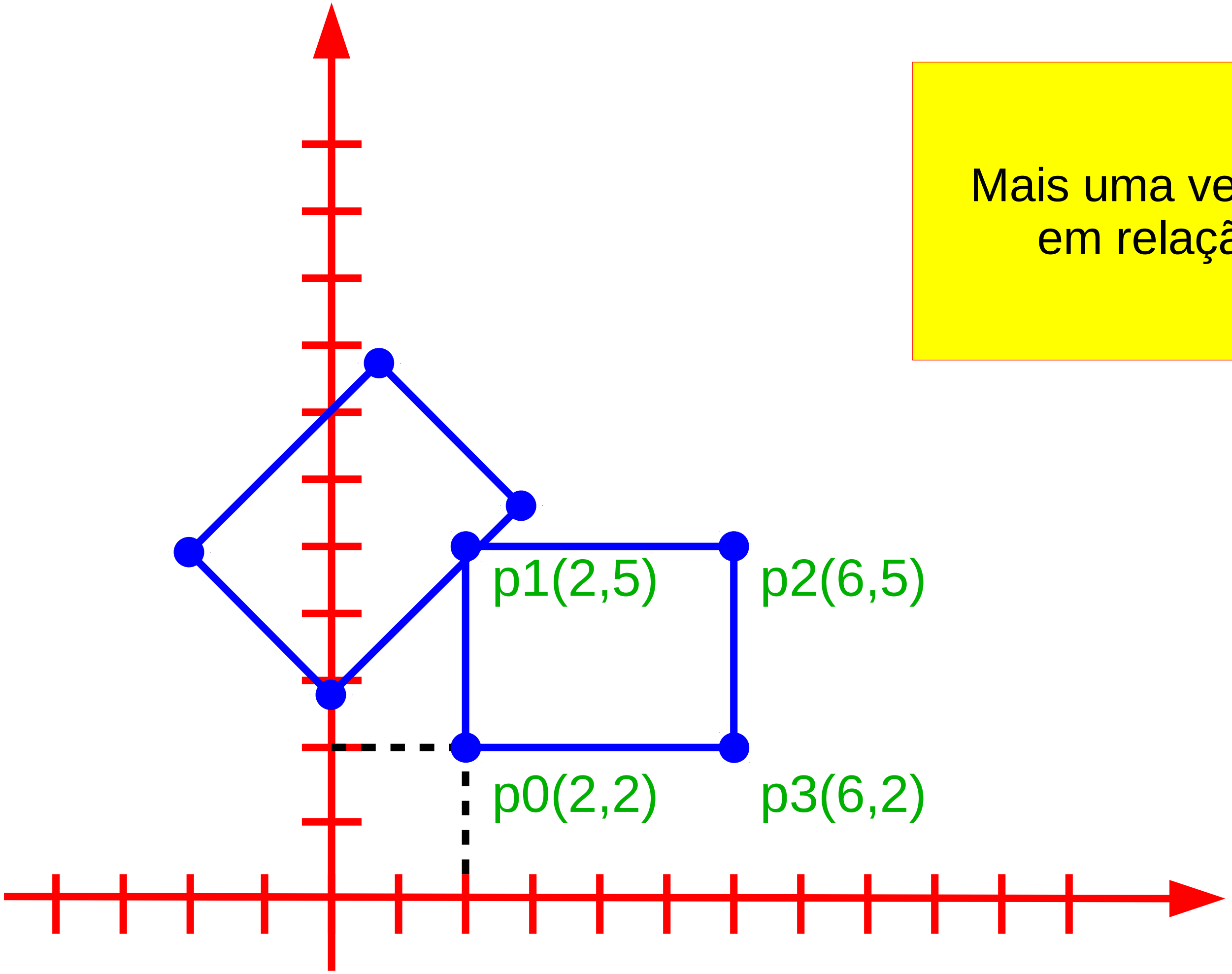
Rotação



Aplica-se a fórmula para **cada** ponto do polígono:

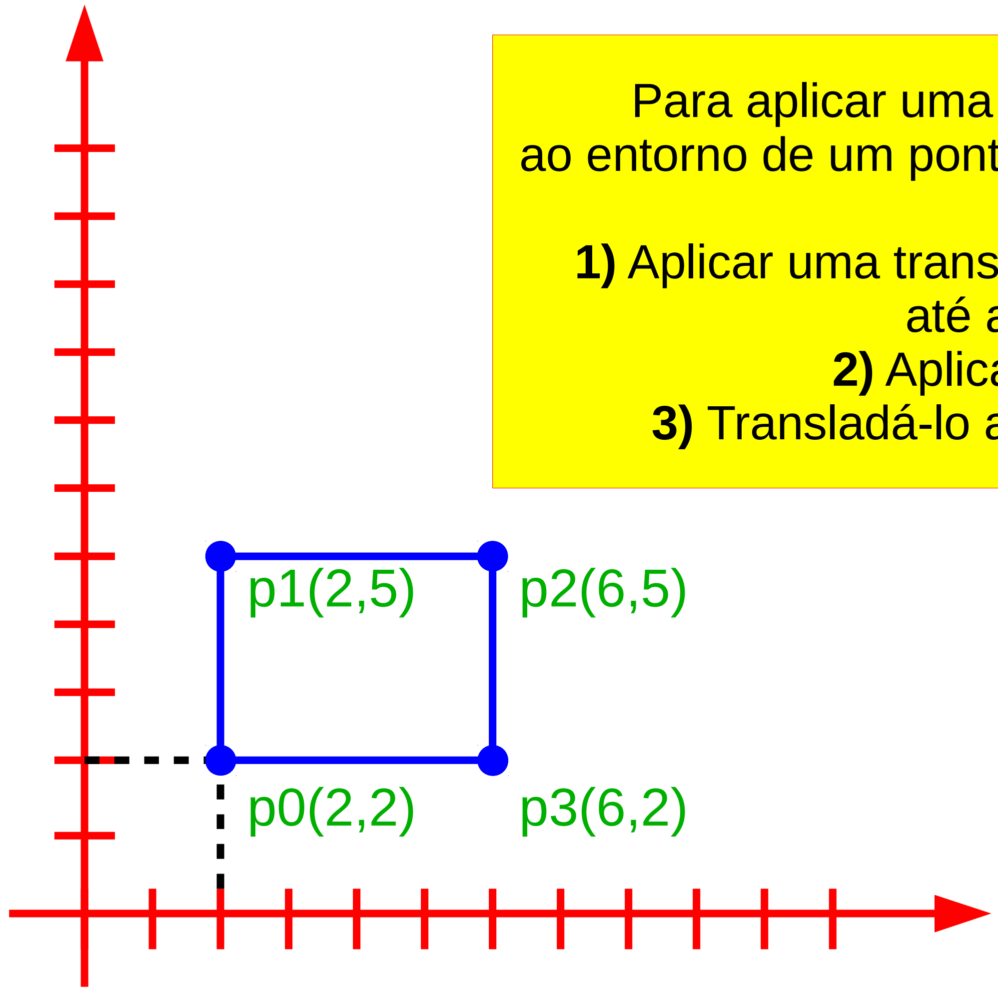
$$x' = x \cos \theta - y \sin \theta$$
$$y' = x \sin \theta + y \cos \theta$$

Rotação



Mais uma vez, a rotação ocorre em relação à origem **(0,0)**

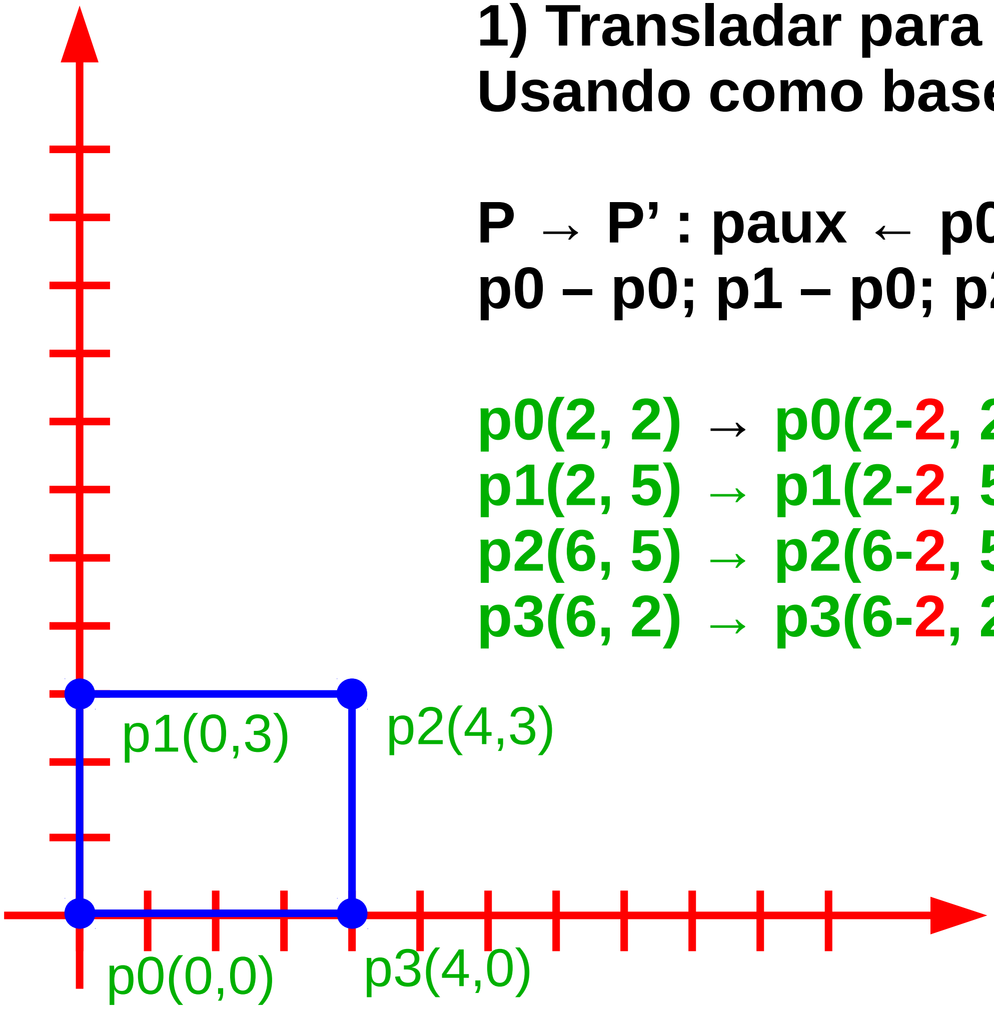
Rotação em torno de ponto árbitrário



Para aplicar uma operação de rotação,
ao entorno de um ponto arbitrário **p**, é necessário:

1) Aplicar uma translação do ponto escolhido até a origem,
2) Aplicar a rotação,
3) Transladá-lo até sua posição inicial

Rotação (Passo 1)

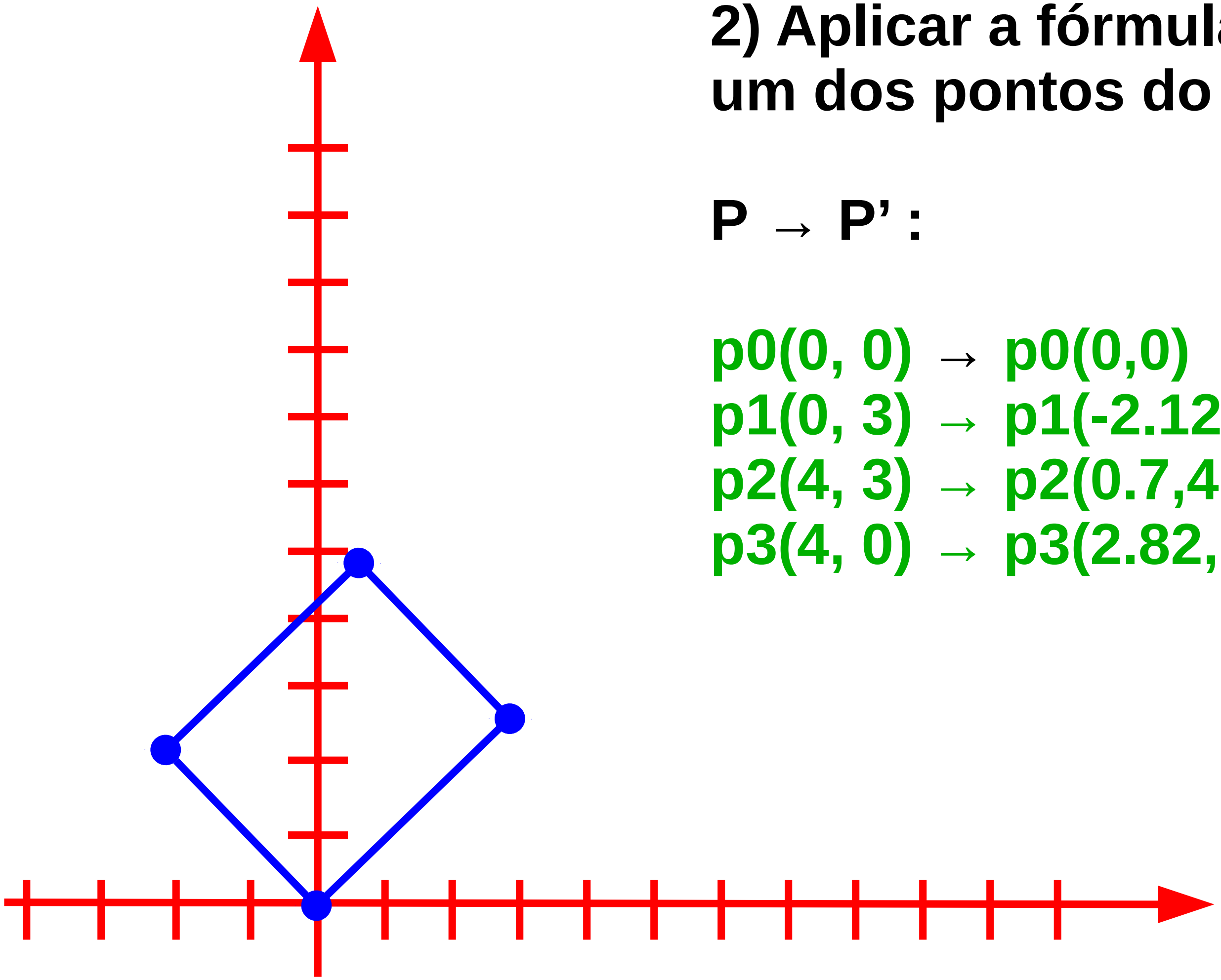


1) Transladar para origem,
Usando como base o ponto p0(2,2):

P → P' : paux ← p0 → paux(2,2)
p0 – p0; p1 – p0; p2 – p0; p3 -p0

p0(2, 2) → p0(2-2, 2-2) = p0(0,0)
p1(2, 5) → p1(2-2, 5-2) = p1(0,3)
p2(6, 5) → p2(6-2, 5-2) = p2(4,3)
p3(6, 2) → p3(6-2, 2-2) = p3(4,0)

Rotação (Passo 2)



2) Aplicar a fórmula para cada um dos pontos do polígono:

P → P' :

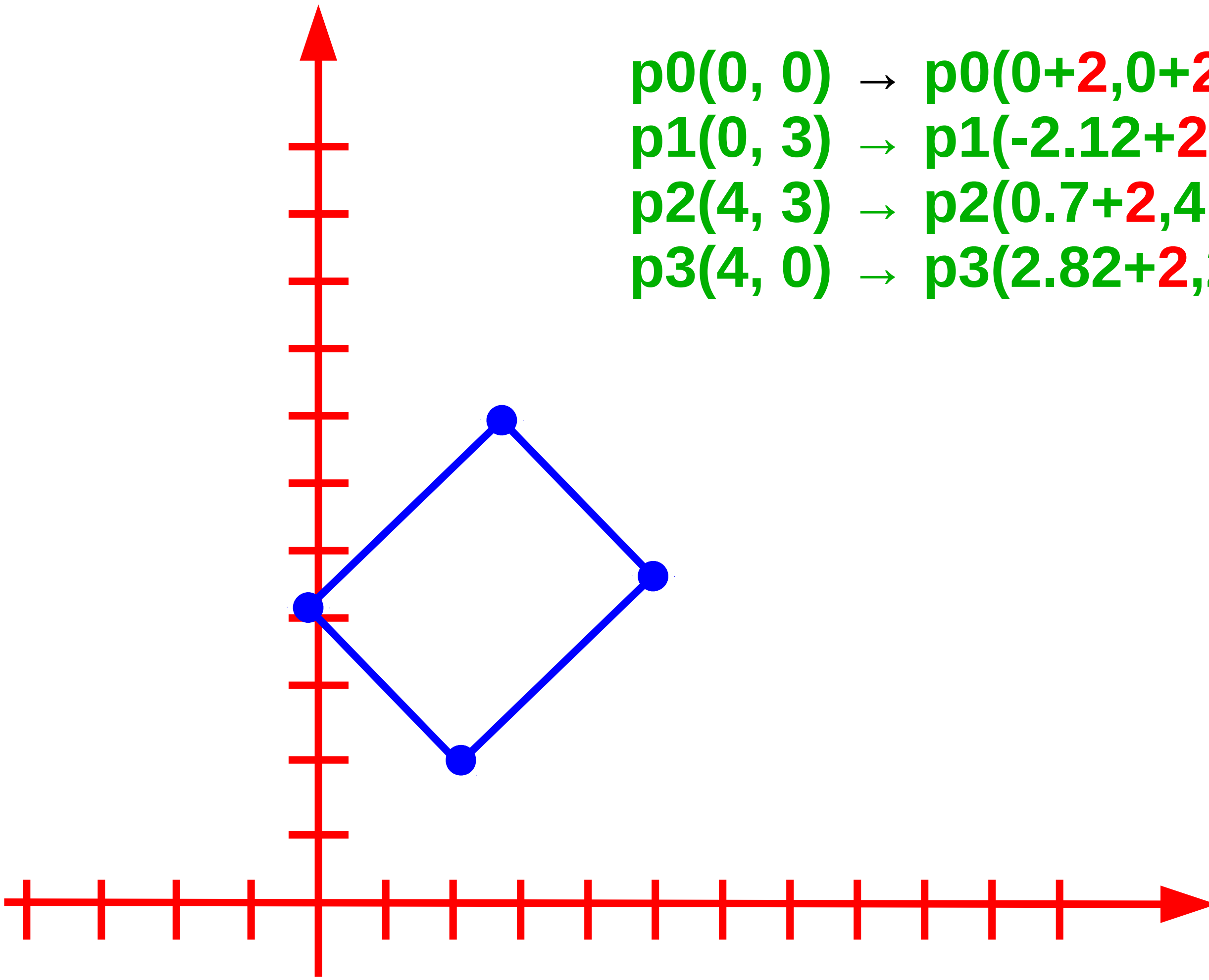
p0(0, 0) → p0(0,0)
p1(0, 3) → p1(-2.12,2.12)
p2(4, 3) → p2(0.7,4.94)
p3(4, 0) → p3(2.82,2.82)

Rotação (Passo 3)

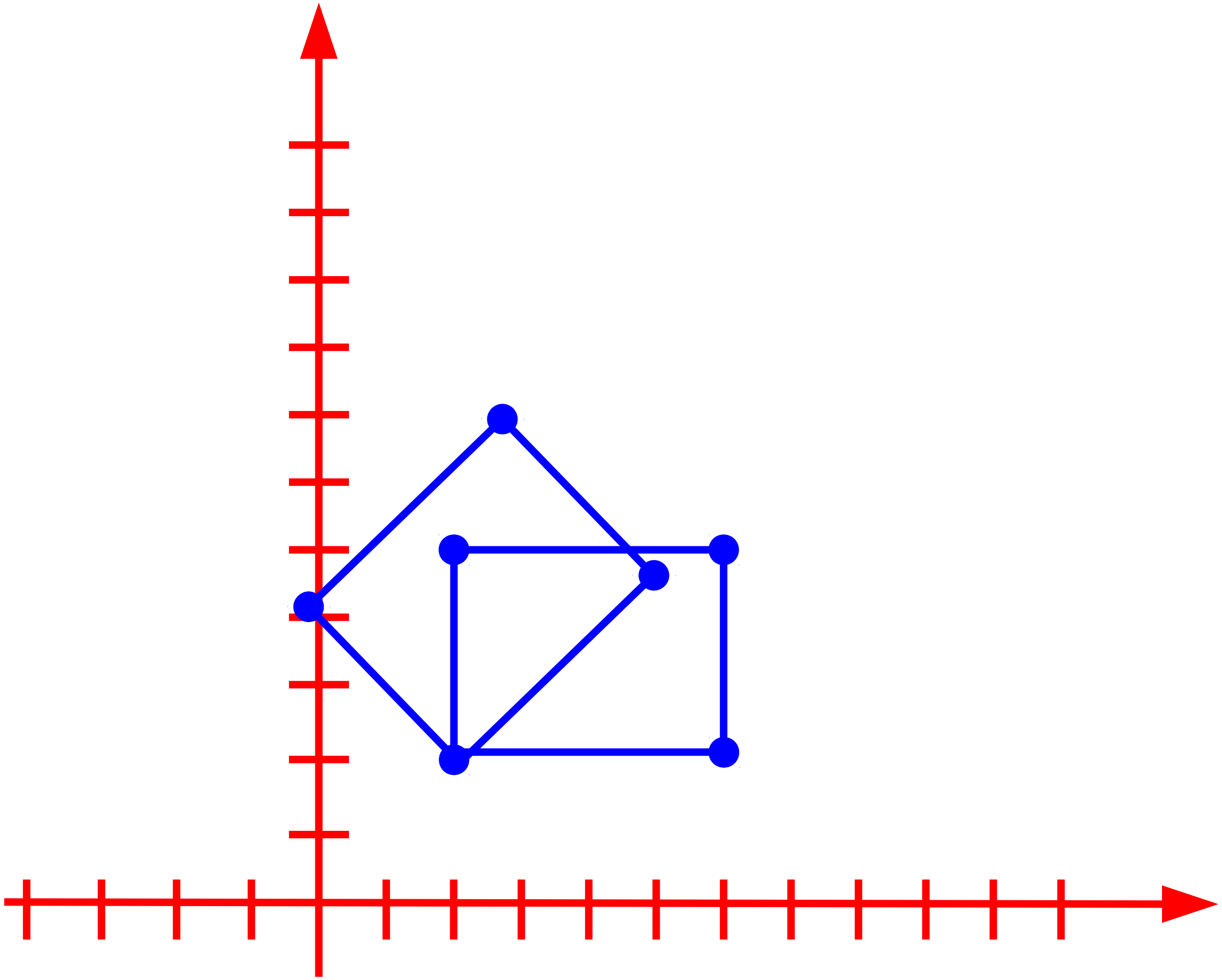
3) Re-adicionar o ponto arbitrário p(2,2) a cada um dos pontos do polígono:

P → P' = P + paux (2,2):

p0(0, 0) → p0(0+2,0+2) = p0(2,2)
p1(0, 3) → p1(-2.12+2,2.12+2) = p1(0.12,4.12)
p2(4, 3) → p2(0.7+2,4.94+2) = p2(2.7,6.97)
p3(4, 0) → p3(2.82+2,2.82+2) = p3(4.82,4.82)



Rotação :
Comparando com o original

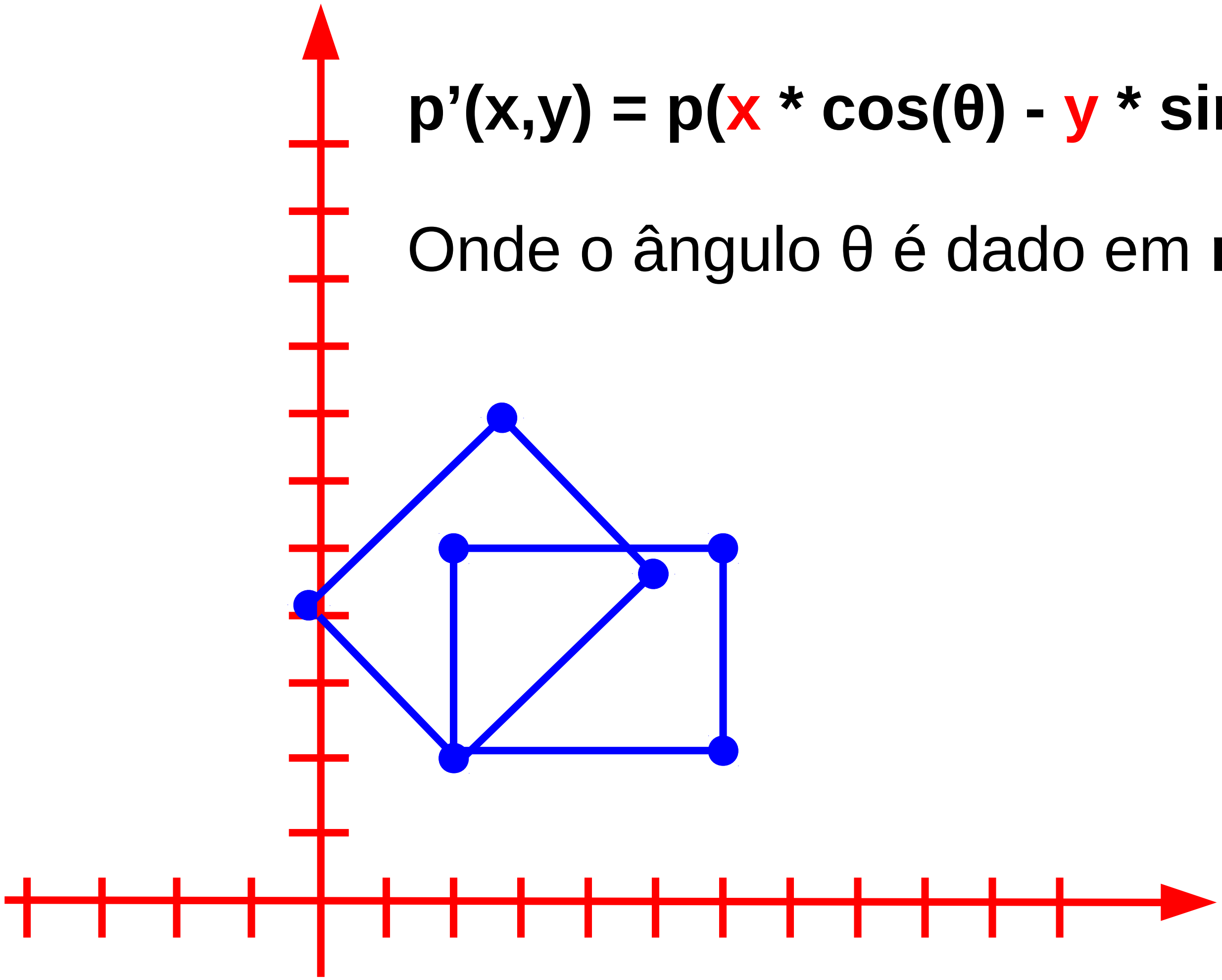


Rotação : Resumindo

P → P’

p’(x,y) = p(x * cos(θ) - y * sin(θ), x * sin(θ) + y * cos(θ))

Onde o ângulo θ é dado em **radianos**

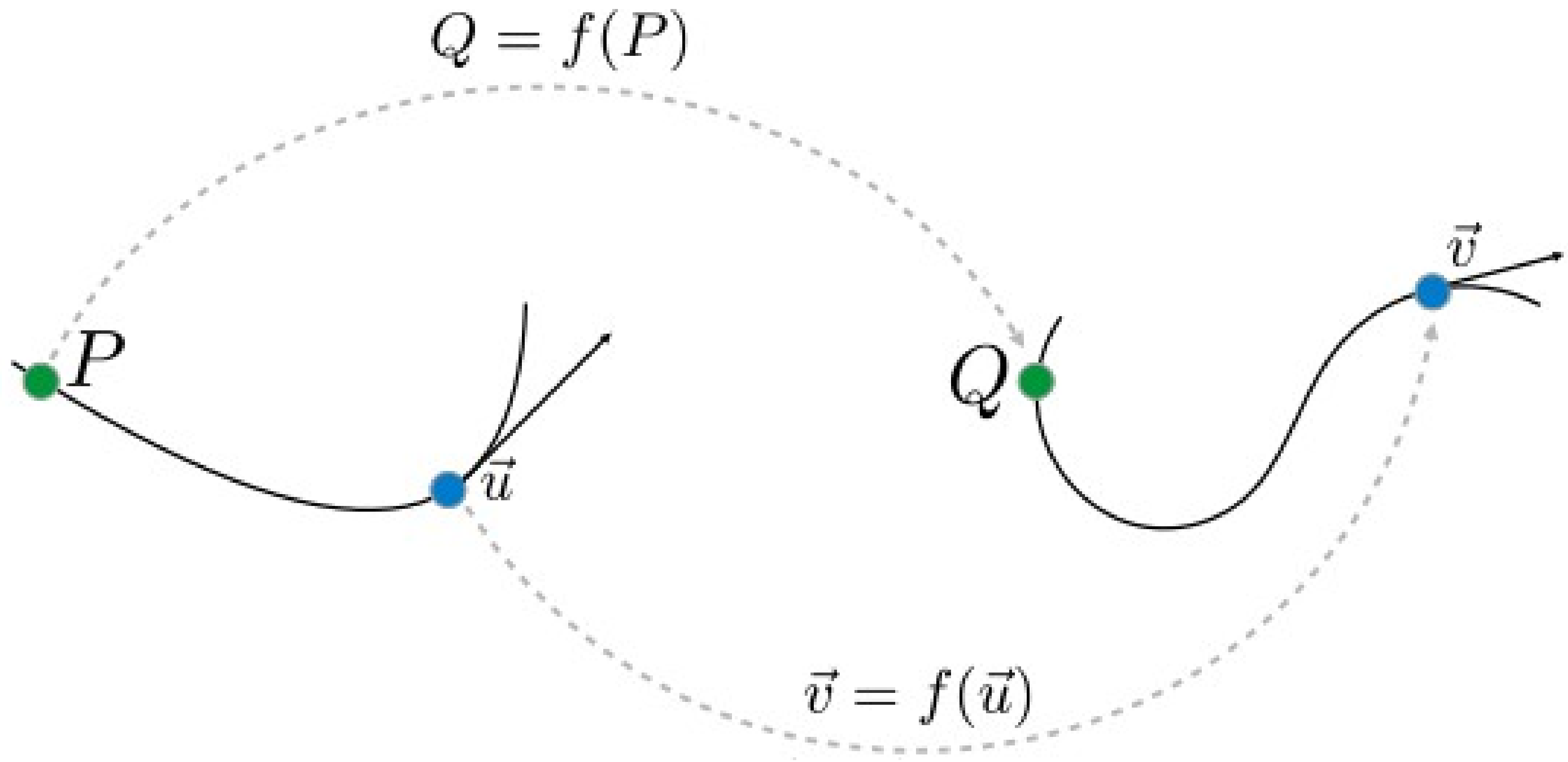


Roteiro

- Transformações Geométricas
- Translação
- Escala
- Rotação
- Transformações Lineares

Transformações Lineares

- Uma transformação geométrica é uma função que mapeia um ponto do espaço em outro ponto.

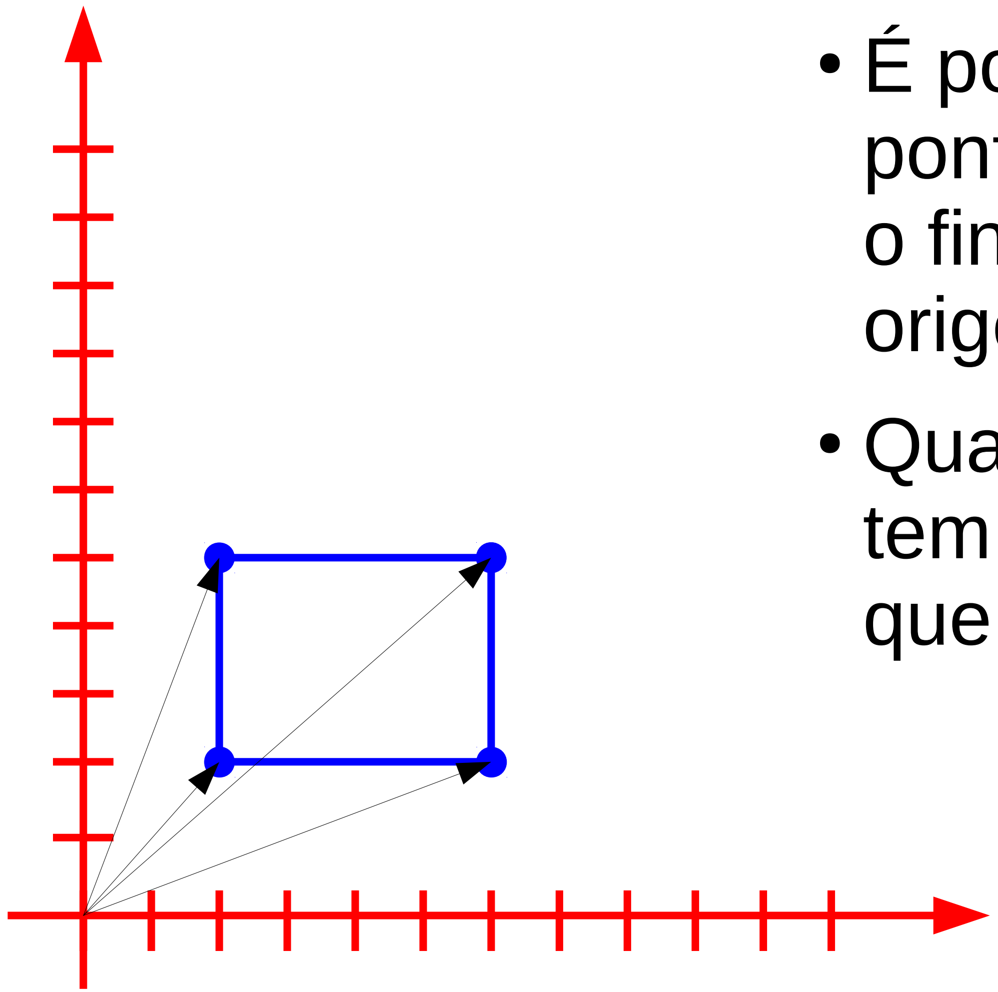


Transformações Lineares

- Uma transformação é dita linear se:

$$f(\alpha P + \beta Q) = \alpha f(P) + \beta f(Q)$$

Coordenadas homogêneas



- É possível considerar cada ponto de um polinômio como o final de um vetor com a origem no ponto (0,0)
- Quando todos os vetores tem a mesma origem, diz-se que o sistema é homogêneo

Transformações Lineares

- Transformações lineares podem ser obtidas através da multiplicação de vetores por matrizes, desde que em um sistema de coordenadas homogêneas.
- Toda transformação linear entre espaços de **dimensão finita** tem forma de matriz

Matrizes de Transformação

- Partindo-se de um ponto (x,y), representado como uma matriz de coordenadas

  Mc = [x y]

- As transformações podem ser escritas como:
  - Translação : Mc + Mt
  - Escala : Mc * Me
  - Rotação Mc * Mr

Matrizes de Transformação

Matriz de Translação

$$Mt = \begin{bmatrix} Tx & Ty \end{bmatrix}$$

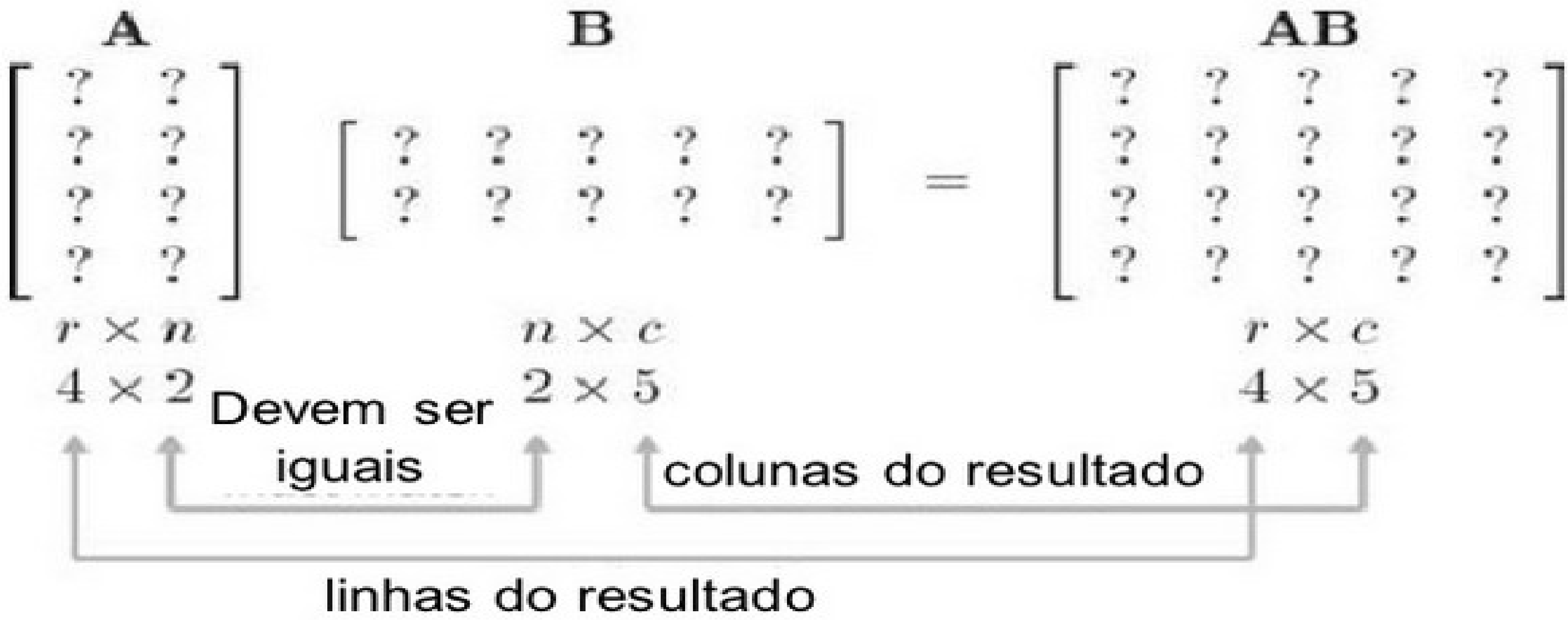
Matriz de Escala

$$Me = \begin{bmatrix} Ex & 0 \\ 0 & Ey \end{bmatrix}$$

Matriz de Rotação

$$Mr = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}$$

Vetores e Matrizes Multiplicação



- Cada termo é o resultado do produto escalar do vetor ***linha i*** com o vetor ***coluna j***

Vetores x Matrizes

- Exemplificando:

$$\mathbf{a} = \begin{bmatrix} 3 & 4 \\ 5 & 6 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$\mathbf{a \times b} = \begin{bmatrix} 3 \times 1 + 4 \times 2 \\ 5 \times 1 + 6 \times 2 \end{bmatrix} = \begin{bmatrix} 3 + 8 \\ 5 + 12 \end{bmatrix} = \begin{bmatrix} 11 \\ 17 \end{bmatrix}$$

Transformações Lineares

- Substituindo os dados do vetor **b** pelas coordenadas x e y de cada ponto, e a matriz **a** pela matriz de transformação desejada, realiza-se a transormação linear de um polígono
- Lembrando-se que essa transformação ocorre sempre em relação ao ponto de origem.

Coordenadas Homogêneas

- Utilizando-se coordenadas homogêneas, é possível realizar as operações de ***Translação, Escala e Rotação*** através de Multiplicações
- Essa abordagem é útil porque permite a combinação das transformações, através de seu produto.

Coordenadas Homogêneas

Matriz de Coordenadas

$$Mc = \begin{bmatrix} x & y & 1 \end{bmatrix}$$

Matriz de Translação

$$Mt = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ Tx & Ty & 1 \end{bmatrix}$$

Matriz de Escala

$$Me = \begin{bmatrix} Ex & 0 & 0 \\ 0 & Ey & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matriz de Rotação

$$Mr = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 1 \\ -\sin(\theta) & \cos(\theta) & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Coordenadas Homogêneas

- Nesse caso, as transformações podem ser expressas por:
  - Translação : Mc * Mt
  - Escala : Mc * Me
  - Rotação : Mc * Mr
- Depois, desconsidera-se o valor obtido na última coluna da matriz resultado

Comutatividade de Transformações

- É importante ressaltar que as **transformações de rotação e escala são comutativas entre si**, ou seja, podem ser aplicadas em qualquer ordem.
- O mesmo **não ocorre** com transformações de translação!

Exemplo de código

- Translação de um polígono :

```
void translatePolygon(Point poly[], int numPoints, double tx, double ty)
{
    for(int i = 0; i < numPoints; i++) {
        poly[i].x = poly[i].x + tx;
        poly[i].y = poly[i].y + ty;
    }
}
```

- Na função ***translatePolygon***:
  - ***poly*** representa um vetor da struct Point, que contém as coordenadas x e y. Ou seja, é o vetor de pontos que compôe um polígono.
  - ***numPoints*** representa a quantidade de pontos contidos no vetor de Points
  - ***tx*** é a quantidade de pontos a transladar no eixo x
  - ***ty*** é a quantidade de pontos a transladar no eixo y

Dúvidas?

Atividade 1

- Aproveitando o projeto **SDL_Clipping**, desenvolvido na aula passada, e sua função de desenho de polígonos:

```
typedef struct {int x, y;} Point;


void drawPolygon(Point pontos[], int qtdPontos, Uint32 cor)
{

    for(int i = 0; i < qtdPontos -1; i++) {
        drawLine(pontos[i].x, pontos[i].y, pontos[i+1].x, pontos[i+1].y, cor);
    }
    drawLine(pontos[qtdPontos-1].x, pontos[qtdPontos-1].y, pontos[0].x, pontos[0].y, cor);

}
```

Atividade 1 (Continuação) :

- Implemente (copie) a função que faz a translação de um polígono
- Desenhe um retângulo de 320 x 240 pixels centralizado na tela, utilizando a função de desenho de polígonos.
- Desenhe um novo retângulo, aplicando uma translação de 20 pixels em x e y do retângulo anterior. Faça isso usando uma cor diferente do primeiro retângulo

Atividade 2 :

- No mesmo projeto, implemente as funções para escalar e rotacionar o polígono, usando como exemplo a função ***translatePolygon***
- Escale o retângulo anterior em 50% (Escala 0.5) e o desenhe novamente, com outra cor
- Rode o retângulo original em 45 graus, desenhando-o com outra cor diferente
- Observe os resultados. Por que houve mudança de posição?

Atividade 3 :

- No mesmo projeto, implemente as funções para escalar e rotacionar o polígono ao redor de um ponto arbitrário
- Novamente escale o retângulo em 0.5, desenhe-o, e depois faça uma rotação do mesmo em 45 graus, agora ao redor do ponto inicial (poly[0]) do polígono.
- Observe os resultados. O que houve com as posições agora?